KIGALI INDEPENDENT UNIVERISTY (ULK)

Machine-learning Prognostic Models and webHealth Application for the 2018–2020 Ebola Outbreak in North-Kivu and Ituri Province

By

KALEMA JOSUE DJAMBA

Enroll**: 202110587**

Supervised By: **Dr. UWITONZE Alfred**

Thesis submitted in Partial Fulfillment for the requirements of the award of the Degree of Master of Science in Internet Systems

AUGUST, 2023

Declaration

I, **KALEMA JOSUE**, hereby declare that this study entitled “ **Machine-learning Prognostic Models and WebHealth Application for the 2018–2020 Ebola Outbreak in North-Kivu and Ituri Province** " submitted in Partial Fulfillment for the Degree of Master of Science in Internet Systems.

I declare that the work reported in this study has not been submitted, for the award of any other degree or diploma in this university or any other learning institution.

KALEMA JOSUE DJAMBA **Reg.no: 202110587**

Date: …...... /............../…………

Signed ………………………

APPROAVAL

This is to certify that the study entitled "**Machine-learning Prognostic Models and WebHealth Application for the 2018–2020 Ebola Outbreak in North-Kivu and Ituri Province"** submitted by KALEMA JOSUE Djamba, Roll number: 202110587 to the Kigali Independent University in partial fulfillment for the Degree of Master of Science in Internet Systems.

Supervisor

Dr. UWITONZE Alfred

Date: …...... /............../…………

Signature ………………………

Dedication

No tribute could ever be adequate to our dear mother **Chantal UCHANGE YANANA's** love, which you never cease to fill. That the All-Powerful grant you good health and a long life**.**

ACKNOWLEDGEMENT

Yahwee, the Almighty, the Universe's Creator and Sustainer, deserves praise.

Throughout the research and writing-up process for my Master, I am grateful to Dr Alfred UWITONZE, who inspired me, encouraged me to carry on despite the odds, and above all read through the work paragraph by paragraph, chapter by chapter, directing me till the end, his patience, guidance, encouragement, and advise during my Master’s studies.

I wish to acknowledge Dr. Olivier HABIMANA , who provided me a good insight for this thesis that helped me to generate knowledge that resulted in the content for this research.

I am grateful thanks to ULB Cooperation and NORTH KIVU's provincial health division for their guidance and allowing me to access and use their recorded data to
make this research successful.

To our entire family, especially our younger brothers, KALEMA DANIEL Jonathan, KALEMA ONAYA Joel, AKAKE SAFI Olive, UWIDUHAYE NDEZE, DJAMBA KALEMA Luc, and YUSUFU KALEMA ,for their moral support, continuous prayers, and endless patience.

The same gratitude is extended to all of our friends and acquaintances.

Abstract

The abrupt explosion of the Ebola virus in 2018 in DRC was one of the world's most widespread and deadliest epidemics with the highest number of casualties being reported in the regions of NORTH KIVU and ITURI provinces. Ebola, a fatal hemorrhagic fever syndrome, is caused by the Ebola virus (EBOV).

The World Health Organization proclaimed the disease as a world healthcare crisis. In most of the cases, the patients are known to have died before the antibodies could respond. This indicates the need to improve upon the diagnosis and prediction techniques available for this disease. This work aims to analyze and improve upon the accuracy of the prediction systems for the Ebola disease using several inputs. The input relies on the symptoms shown by the patient during the early stages of the disease. The data mining techniques employed to carry out this research include Decision Trees, KNN, Support Vector Machine, Random Forest, Gradient Boosting classifier. The experimental results show the accuracy obtained by each classification technique and the best predictive model for both diagnosis and prognosis was Support Vector Classification that were applied to the dataset with 0,88 accuracy.

We will include these models into an Ebola prediction web app with an API in Flask(Python), which will aid medical practitioners and people in the early diagnosis of illness.

Keywords Words: Ebola Virus, Hybrid Neural Network, Decision Tree, KNN, Support Vector Machine, Random, API, Flask

Table of Content

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATION AND ACCRONYMS

ACT:	Artemisinin Combination Therapy
AI	Artificial intelligence
ANN:	Artificial Neural Network
ARIMA:	Auto- Regressive Moving average
AR:	Auto régressive model
CHWs:	Community Health Workers
DRC	Democratic Republic of Congo
EMR:	Electronic Medical Records
FP	False Positive
FP:	True Negative
FN:	False Negative
HMIS:	Health Management Information System
IRS:	Indoor residual Spraying
KNN:	K-Nearest Neighbour
NB:	Naïve Bayes
PERT	Program Evaluation Review Technique
RBC:	Rwanda Biomedical Centre
ROC:	Receiver Operating Characteristic
SDG:	Sustainable Development Goals
SVM:	Support Vector Machine
TP:	True Positive
UML	Unified Modeling Language
WHO	World Health Organization

Chapter I: General introduction

This chapter introduces the idea on Ebola it goes on historical background of Ebola and various interventions have been made to eradicate Ebola in endemic regions, it shows the Ebola picture globally, regionally and its distribution in DRC. Moreover, this chapter includes statement of the problem of Ebola and shows the contribution of the technology in prevention and control of Ebola outbreak, it talks about the research objectives, scope of the study, significance of the study and organization of the study which shows the flow of the study.

1.1 Background of the study

Ebola Virus Disease (EVD) is a sort of a disease caused by infection with a virus in the group of the Filoviridae family.

In 1976, EVD first appeared in two covering flare-ups, one in what is presently Nzara, South Sudan, and the other in Yambuku, DRC. The last happened in a town close to the Ebola River, from which the disease takes its name (World Health Organization, 2022).

The disease has infected humans on occasion, resulting in outbreaks in many African countries including the latest recent in August 2018 in the Democratic Republic of the Congo (Situation on the current outbreak in North Kivu, 2022).

On May 8, 2018, the World Health Organization (WHO) announced the occurrence of an outbreak of Ebola virus disease (EVD) in the Democratic Republic of Congo (DRC) (disease, 2023).

Between April 4 and May 7, 21 suspected EVD cases were reported in the Équateur Province towns of Iboko and Bikoro. On May 7 , blood samples from five hospitalized patients were transported to Kinshasa for Ebola-PCR testing on May 7, with two confirmed PCR-positive. (Organization response, 2018).

The immunization of healthcare personnel began on May 21 (H, 2018). By May 27, the ring vaccination effort was in full swing, with 906 contacts and contacts of contacts being actively tracked. Six suspected, thirteen probable, and thi`rty-five confirmed EVD cases had been reported, with 25 (52%) of 48 probable and confirmed EVD cases dying (disease, 2023).

This outbreak had various characteristics that were concerning for extensive transmission. Cases were recorded throughout a 168-kilometer radius, including four confirmed cases in Mbandaka, the 1,200,000-person province capital of Equateur, which is located on the Congo River and borders Congo.-Brazzaville, (Organization, 2022).

Furthermore, flights from Mbandaka to Kinshasa are regular. Given these risk variables, early epidemic growth profiles, (Chowell, L, Bansal S, & Viboud C, 2016) and evidence of previously unreported infection from prior outbreaks, the risk of a significantly bigger outbreak could not be dismissed (JD, Barrie MB, Mesman AW, & al., 2018) .

The factors causing epidemic growth to peak have been debated. Delayed detection of EVD outbreaks and resulting widespread distributions of EVD have significantly contributed to epidemic growth. (Team, 2016) In addition to traditional burial practices, Ebola treatment units with low quality care and/or high mortality rates have discouraged Ebola suspects from presenting to care and contribute to community-based transmission.

Fragile, overburdened public health surveillance systems have also contributed to increased rates of unreported cases, putting urban groups at risk, which may have had higher transmission rates than rural ones, (BD, MSY, Tiffany A. al, 2018).

According to (F, Gsteiger S, Low N, Hansen CH, & Althaus CL, 2016) A shift to subcritical transmission (reproduction number below 1) occurs when Ebola response groups implement control, prevention, and care measures communities adopt more protective behaviors, and/or transmission in a social network reduces (S, Ciglenecki I, Tiffany A, & al, 2017) (ME, 2017).

Scientific advances with rapid diagnostics and vaccines from the West Africa outbreak were deployed in the April-July 2018 EVD outbreak in DRC and had the potential to limit Ebola virus transmission.

There have been 3,348 confirmed cases of Ebola infection, including 2,210 deaths.

The disease is now being identified as one of the primary reasons of Africa's recurrent conflicts, socioeconomic stagnation, and decreasing development. (Brown & Johnson, 2018).

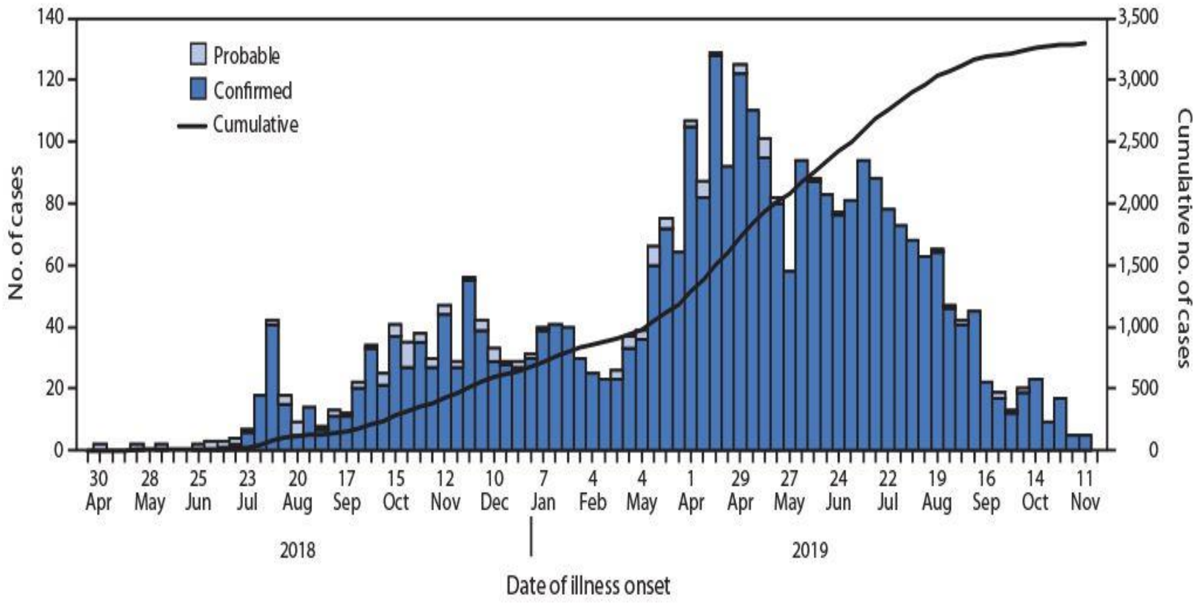


Figure 1:statistics of confirmed cases, deaths from the ebola virus between May 2018 and November 2019 Source: (Centers for Disease Control and Prevention,2023)

Machine learning is utilized in many disciplines, and users are sometimes unaware that they are doing so. Machine learning is used in a variety of domains, including recognizing faces, speech recognition, disease prediction, self-driving cars, web search, and anomaly detection, among others. Artificial intelligence (AI) has begun to revolutionize healthcare in industrialized markets and has the potential to generate game-changing changes in global health for areas of disadvantage. From enabling community health workers to better serve patients in remote areas to helping governments in low and middle-income countries (LMICs) in preventing deadly disease outbreaks before they occur, there is growing recognition of AI tools' tremendous potential to break fundamental trade-offs in health access, quality, and cost (M. M. , 2020).

Furthermore, machine learning is commonly utilized in assessing health, diagnosing diseases such as cancer, and predicting illness incidence. Machine learning is used in pharmacology to find the proper formula and dependable medications to cure a certain ailment, and it is widely employed in selecting the most successful therapeutic therapy. Among machine learning techniques, artificial neural networks (ANN) have proven to be very powerful due to the black box and volatile learner concepts, and deep learning has experienced exponential growth due to the ability of insightful decision making, resulting in high-level abstraction in data.

Deep learning also provides an improved performance when dealing with irregular and non-stationary time series data because it discovers and characterize the complex features of data set and back-propagation of neural network is one of the most used methods in deep learning. However, decision trees have high ability in discovery, in accuracy, preciseness and reliability (O & G. Jimoh R, 2018).

Furthermore, one application of machine learning in health care employed for the processing of big and complicated datasets so that clinical insights uncovered through health data, the implementation of machine learning in health care also led to an increase in patient satisfaction. Furthermore, machine learning is widely employed in disease prediction, with disease prediction made feasible by utilizing machine learning predictive algorithms, resulting in smarter health care. As a result, machine prediction of disease and epidemic outbreaks leads to the early application of preventive measures.

According to (Yan, & al., 2018) ,Data mining is critical for extracting deep insights from massive datasets.At the moment, the rising usage of data mining techniques in industries such as healthcare has altered the approach taken by working experts to solve an issue. The publicly available laboratory and clinical databases include numerous tests required to diagnose a certain condition.

Data mining has evolved into an unavoidable component of any research.This is related to the massive volume of data generated globally.

The increased knowledge of disease prediction models has prompted academics to apply existing prediction and classification approaches to various disease databases (Kotsiantis, Zaharakis, I.,, & Pintelas, P., 2017).

Attempts have been made to improve disease diagnosis through the development of new and more efficient algorithms. The Naive Bayes classifier, which is based on Bayes' Theorem, is one of the most extensively used approaches (Dudzik, 2018).

It is assumed in this classifier that the existence of any single feature set is unrelated to any other feature.

The features are known to contribute to the likelihood independently. This model demonstrates its efficacy in terms of simplicity, efficiency, and rapid prediction for any dataset.The SVM classifier is another classification technique that is used to evaluate data for regression and classification.It is a supervised learning model that separates objects with distinct class memberships using a decision plane.

SVM algorithm proves to be efficient when applied to complex problems such as text and image classification,etc. Other widely used techniques include Decision Tree, Bagging Classifier, Random Forest and so on.

1.2 Problem Description

The Ebola virus disease was first detected in the Democratic Republic of the Congo in 1976.From 1976 to the present, ten outbreaks have been reported.

The most recent outbreaks have occurred in the Democratic Republic of the Congo. The Democratic Republic of the Congo's Ministry of Health declared an official outbreak of the Ebola Virus Disease (EVD) in the North Kivu area, a hilly terrain in the country's upper east, on August 1, 2018. The outbreak's epicenter was tracked to the territory surrounding Beni (US, 2019).

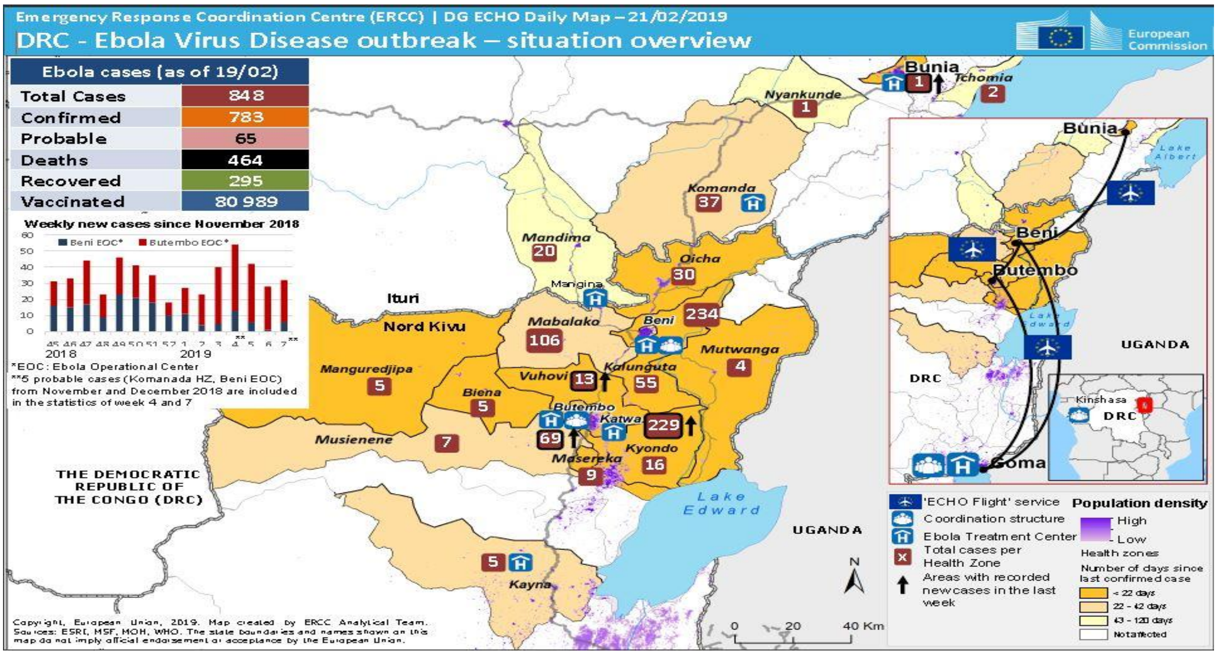
As of 4 December 2019, more than 3 300 cases had been confirmed in the Democratic Republic of the Congo, with over 2 200 recorded Ebola deaths. More than 223 000 Congolese have received the rVSV-ZEBOVGP Ebola vaccine (SEMAJERI & UMEZURUIKE CHINECHEREM, november 2019).

WHO also reports the usage of two Ebola medications in the DRC as part of a clinical trial that has been proved to save 9 out of 10 lives when administered at the appropriate time (US, 2019).

The Government of DRC is collaborating with different Health organization for combating against this disease by reducing the numbers of death caused by Ebola . When this disease is not discovered at its earlier stage, it can put the life of many people at risk and even reduce the workforce of the country. Thus, there is still a need to use the ICT technology to help this initiative to access on the information relating to the development of Ebola on time before this disease spread out across the citizen.

The recent Ebola virus disease (EVD) outbreaks have highlighted the necessity for field-deployable patient management technologies that can adapt to the disease's broad spectrum of pathology across a wide range of locations and resources.

Given the disease's reemergence in December 2019, there is an urgent need to study its course and use the existing available patient information to construct multiple disease forecasting models. Such models, which can detect signs at the outset of an illness, would help healthcare administrators and staff provide better and more timely healthcare (Goss & Ramchandani, 2018).



Ebola cases (as of 19/02)	
Total Cases	848
Confirmed	783
Probable	65
Deaths	464
Recovered	295
Vaccinated	80 989

Emergency Response Coordination Centre (ERCC) | DG ECHO Daily Map – 21/02/2019
DRC - Ebola Virus Disease outbreak – situation overview
European Commission
Weekly new cases since November 2018
Beni EOC*
Butembo EOC*
60
50
40
30
20
10
0
45 46 47 48 49 50 51 52 1 2 3 4** 5 6 7**
2018
2019
*EOC: Ebola Operational Center
**5 probable cases (Komanada HZ, Beni EOC) from November and December 2018 are included in the statistics of week 4 and 7
Bunia
1
Tchomia
2
Nyankunde
1
Komanda
37
Mandima
20
Oicha
30
Mangina
Ituri
Nord Kivu
Mabalako
106
Beni
234
Mutwanga
4
Manguredjipa
5
Vuhovi
13
Kalunguta
55
Biena
5
Butembo
Katwa
229
69
Musienene
7
Kyondo
16
Masereka
9
THE DEMOCRATIC REPUBLIC OF THE CONGO (DRC)
Lake Edward
UGANDA
5
Kayna
Lake Albert
DRC
Kinshasa
Goma
N
0 20 40 Km
'ECHO Flight' service
Coordination structure
Ebola Treatment Center
Total cases per Health Zone
Areas with recorded new cases in the last week
Population density
High
Low
Health zones
Number of days since last confirmed case
< 22 days
22 - 42 days
43 - 120 days
Not affected
Copyright, European Union, 2019. Map created by ERCC Analytical Team. Sources: ESRI, MSF, MOH, WHO. The state boundaries and names shown on this map do not imply official endorsement or acceptance by the European Union.

Figure 2 : DRC - Ebola Virus Disease outbreak – situation overview DG ECHO Daily Map | 21/02/2019

Source: (Centers for Disease Control and Prevention , 2023)

1.3 Study Objectives

1.3.1 General Objective

The main objective of this paper is to employ the Machine Learning for Ebola outbreak prediction in DRC specifically in NORTH KIVU and ITURI provinces to make a web application using Flask for python.

1.3.2 Specifics Objectives

The main contributions of this paper are as follows:

i. To develop hybrid models to improve disease detection.

ii. To proposed extremely helpful to medical practitioners and laboratory professionals in the early detection of sickness, potentially saving patients' lives.

iii. To study analyses existing models and compares them to the proposed hybrid models.

iv. To develop an Ebola prediction web app with an API in Flask, which will aid medical practitioners and people in the early diagnosis of illness.

1.4 Research Questions

i. How to develop hybrid model?

ii. How to study the current model?

iii. How to prepare extremely helpful model using Machine learning?

iv. How to develop API using ML algorithm?

1.5 Scope

This research Dissertation aims to construct a predictive machine learning on Ebola outbreak in DRC especially in Noth Kivu and Ituri Province. I will experiment with dataset made public by the ULB Cooperation in partnerships with NORTH KIVU's provincial health division.

This study is based on the largest and most diverse clinical EVD dataset available to date from 2018 to 2020, comprising 28,904 records, which comprise 2,756 verified cases, 15,774 unconfirmed cases, 10,480 suspected cases, and 94 probable cases from five different locations in DRC / NORTH KIVU. That is how the study will be limited both in location, time and size.

We propose a hybrid predictive model (G., Varatharajan R., & Priyan M. K., 2018) to fulfill our objective of the Ebola disease prognosis. In doing so, various classification techniques (V., 2016) were used with neural networks to develop hybrid models. A comparative study has been made to better understand the results (P. & Ramchandani H., 2018). We have made all of the resources employed in this project publicly available to facilitate further research, development and improved initial responses in case of an outbreak.

1.6 Significance of the Study

Nowadays, there is an increase in the use of computer-based technology, such as machine learning, in the health sector, such as in the diagnosis and treatment of diseases such as cancer. Machine learning also plays an important role in the prediction of risk factors associated with the presence of certain diseases. This study enables ebola outbreak prediction using machine learning techniques and the best algorithm chosen for ebola outbreak prediction based on accuracy and other performance criteria. This study assists health care providers, hospitals, and health organizations in anticipating the emergence of an ebola outbreak so that they can take additional precautions and control measures to preserve people's lives.

1.7 Summarized approach

Methods

- ***Supervised Learning***

Supervised learning is a form of machine learning method in which we feed sample labeled data to the machine learning system in order to train it, and it predicts the output based on that.

The system builds a model using labeled data to interpret the datasets and learn about each data. After training and processing, we test the model by supplying sample data to see if it predicts the precise output or not.

The purpose of supervised learning is to connect input and output data. Supervised learning is dependent on monitoring, such like when a student learns something under the observation of a teacher. Spam filtering is one example of supervised learning (Munappy, Jan Bosch, Helena Holmström Olsson, & Anders Arpteg, september 2022).

- ***Agile development method:*** Assists customers in refining or agreeing on a system design while ensuring everyone is satisfied with the ultimate outcome. Using unified modeling language (UML) diagrams to extend this visual approach to the agile development starting point might help deliver real outcomes for a project (Fenn, 2023).

1.8 Organization of the Study

This study is made by five chapters. Chapter one contains the introduction, the problem statement, objectives of the study, scope, and significance of the study. Chapter two is made by the literature on prediction of ebola using machine learning and the basic concepts of machine learning. Chapter three assess the methodology used to extract data, train, and build model and evaluation of machine learning algorithms used in the study. Chapter four presents the results; analyses building of machine learning algorithms and discussion of the findings got using the stated methodology. Chapter five concludes the study and gives the recommendations to focus on based on findings got.

Chapter II: LITERATURE REVIEW

2.1 Introduction to the Review

This segment gives the state of earth through eyes of previous studies that this study builds on. It also reviews the various concepts under study and highlights on the gap in past studies that this study seeks to fill. It further lays the theoretical framework upon which this study is built on.

2.2 Previous studies in the area under study

Innumerous studies have been published which focus on Ebola disease prognosis using various classification algorithms and machine learning models. Various attempts have been made to understand the Ebola Virus and classify it to help develop better healthcare decision making among healthcare professionals.

According to (S. & Mangat, september 2017) , The primary goal was to apply data mining techniques to the Ebola Disease Virus dataset in order to classify the disease and create a comparative study between it and other epidemic diseases.

They present a work that generalizes error and intraclass separability. The relevance vector machine classifier is used to do this. The authors of this paper classified Ebola virus data on the basis of its distribution throughout multiple continents.

The RVM classifier was run after several elements such as RVM weight and bias data, testing feature vector, and group data were submitted.The decision reasoning was returned after evaluating the corresponding RVM categorization information.

In 2016, Andres Colubri, Tom Silver and the other authors (A., et al., 2018), By evaluating the earliest symptoms displayed by the patient's body, a machine-learning-based structure and self-developed software were used to forecast the health status of Ebola patients.

They investigated the issues created by insufficient clinical data. Realizing the demand for clinical prognostic mobile apps, the app shown the generation of actionable knowledge from systematic data collecting in order to improve clinical, laboratory decision making among clinical, and laboratory professionals.

Kanika Chuchra and Amit Chhabra employed tree-based mining algorithms to Ebola Virus Dataset. Filtering the dataset to remove noise from the dataset improved the results even further. In addition, the authors used the J48, LMT, and REP algorithms. To achieve better outcomes, an unsupervised filter was used in conjunction with various algorithms. WEKA and MATLAB were utilized as tools.

The testing findings showed that using the LMT classifier in conjunction with the Random tree produced better results, with an accuracy rate of 98.3193% (K. & Chhabra A., 2019 September).

M. Jana Broadhurst Tim J. G. Brooks and Nira R. Pollock described the progress and recent developments in the diagnostic testing of the Ebola disease virus.

They also looked into the procedures taken to set up diagnostic facilities in the area where the disease was spreading. Furthermore, they investigated the difficulties encountered during the various stages of on-site diagnosis in order to give a thorough evaluation of the numerous diagnostic tests used to address the issue up to this point (J., Brooks T. J., & Pollock N. R., 2018).

Manu Anantpadma, Thomas Lane and various other authors elaborated

upon the previous Bayesian machine learning models which were approved from the FDA and were employed for the identification of various compounds that are active against the Ebola virus.

The levels of tilorone (one of the active compounds) were used to make conclusions when the active molecules were identified. The application of current models, together with their chemical knowledge, resulted in a novel strategy for prioritizing compounds for in vitro testing. The study went on to investigate the possibility of extending such improved models and approaches to other diseases (M., et al., 2019).

In 2015, Peng Zhang, Bin Chen, Liang Ma and various other authors (Zhang, et al., 2017) emphasized that the accuracy and reliability of various experimental outcomes could be studied better with the aid of artificial society.

They demonstrated the construction of artificial Beijing and the Ebola propagation model according to the conditions in West Africa. Further, the propagation nature of the virus along with epidemic conditions was analyzed and corresponding results were presented. The study concluded that the Ebola outbreak is impossible to occur in the city of Beijing.

In (Colubri, et al., 2019) , authors described a work, which is based on the largest and most diverse clinical EVD dataset to date, which includes 470 confirmed EVD cases from five distinct locations in Sierra Leone and Liberia, as well as 264 cases from two independent datasets for external validation.

It demonstrates how data harmonization approaches can provide interoperability between heterogeneous datasets and builds a family of adaptable externally validated prognostic models capable of approximating observational wellness assessments made by trained doctors. We then incorporate these models into the first field-deployable smartphone app for EVD prognosis, which provides informed access to recommended guidelines.

2.2.1 Critical analysis of existing related works

All of our related works prediction models were constructed utilizing data collected over a period of time and for people living together and sharing the same culture, climate, and so on.

Also, these authors are all limited to presenting the best model for predicting Ebola diseases for a given region and based on clinical characteristics, but they have not been able to generate a model based on symptoms observed at the start of the disease. Finally, these authors have not been able to deploy this model in an application, so that anyone can use these models simply by accessing a web or mobile application.

2.3 What distinguishes this study from previous studies: Added value our study

Most of all previous Ebola Prediction models predict the probability of Ebola spread on a community.

The proposed Ebola Prediction model predicts the probability of an individual having Ebola virus based on his / her symptoms.

This is accomplished with the help of Machine learning algorithms and the past available Datasets which are used to train the classifiers for the better predictions.

Following are the few conclusions that we can take from this research:

- This study is based on the largest and most diverse clinical EVD dataset available to date, comprising 28 904 records, which comprise 2 756 verified cases, 15 774 unconfirmed cases, 10 480 suspected cases, and 94 probable cases from five different locations in DRC/NORTH KIVU AND ITURI Provinces (Beni,Bunia,Butembo,Goma, Komanda, Mangina, Tshomia).
- This work aims to analyze and improve upon the accuracy of the prediction systems for the Ebola disease using several inputs. The input relies on the symptoms shown by the patient during the early stages of the disease. The data mining techniques employed to carry out this research include Decision Trees; Bagging classifier, KNN, Support Vector Machine, Stochastic Gradient Descent classifier, Random Forest, Gradient Boosting classifier, Ridge Classifier. The experimental results show the

accuracy obtained by each classification technique and the hybrid models that were applied to our dataset.

- We further integrate these models in the deployable a web app for EVD prognostication, which enables informed about predictions.

2.2 . THE THEORETICAL FRAMEWORK OF THE STUDY

2.2.1 What is artificial intelligence (AI)?

Artificial intelligence is the simulation of human intelligence processes by machines, especially computer systems. Specific applications of AI include expert systems, natural language processing, speech recognition and machine vision (Burns, Nicole Laskowski, & Linda Tucci,, 2023).

Figure 3 : Présentation of AI.   Source : *(javatpoint, 2023)*

2.2.2 How does AI work?

As the excitement surrounding AI has grown, manufacturers have been scrambling to showcase how AI is used in their products and services. What they call AI is frequently only a component of technology, such as machine learning. AI necessitates the use of specialized hardware and software to write and train machine learning algorithms. There is no particular programming language that is synonymous with AI, however Python, R, Java, C++, and Julia all contain characteristics that are popular among AI engineers (Burns, Nicole Laskowski, & Linda Tucci,, 2023).

In general, AI systems operate by consuming huge volumes of labeled training data, analyzing the data for correlations and patterns, and then using these patterns to forecast future states. By examining millions of instances, a chatbot fed text samples can learn to make life like dialogues with people, while an image recognition program can learn to recognize and describe items in photographs. New generative AI algorithms that are quickly improving can generate realistic text, graphics, music, and other material.

AI programming focuses on cognitive abilities such as the following (Goldman, 2013):

- **Learning.** This aspect of AI programming focuses on acquiring data and creating rules for how to turn it into actionable information. The rules, which are called algorithms, provide computing devices with step-by-step instructions for how to complete a specific task.
- **Reasoning**. This aspect of AI programming focuses on choosing the right algorithm to reach a desired outcome.
- **Self-correction**. This aspect of AI programming is designed to continually fine-tune algorithms and ensure they provide the most accurate results possible.
- **Creativity**. This aspect of AI uses neural networks, rules-based systems, statistical methods and other AI techniques to generate new images, new text, new music and new ideas

2.2.3 Branches of Artificial Intelligence

AI research has successfully produced useful approaches for a wide range of challenges, ranging from game play to medical diagnosis.

There are numerous branches of artificial intelligence, each with its own specialty and set of methodologies. Some of the most important branches of AI are (Biswal, 2023):

- **Machine learning:** is concerned with the development of algorithms that can learn from data. Image identification, spam filtering, and natural language processing are all applications that use ML algorithms.
- **Deep learning:** It is a branch of machine learning that harnesses artificial neural networks to acquire knowledge from data. Deep learning algorithms effectively solve various problems, including NLP, image recognition and speech recognition
- **Natural language processing (NLP):** is concerned with the interplay between computers and human language. NLP techniques are utilized in a variety of applications, including machine translation, speech recognition, and text analysis, to interpret and process human language.
- **Robotics:** It is a field of engineering that deals with robot design, construction, and operation. Robots can perform tasks automatically in various industries, including manufacturing, healthcare, and transportation (Biswal, 2023).
- **Expert systems:** They are computer programs designed to mimic human experts' reasoning and decision-making abilities. Expert systems are used in various applications, including medical diagnosis, financial planning, and customer service (Biswal, 2023).

2.2.4 What is an Artificial Intelligent model?

An AI model is a mathematical model that is used to anticipate or make judgments. Some examples of common AI models include:

- Linear regression
- Logistic regression
- Decision trees
- Neural networks

2.2.5 What are the 2 categories of AI?

- **Weak AI:** A sort of artificial intelligence that can only execute specified tasks. A poor AI, for example, might be able to play chess or translate languages (Escott, 2023).
- **Strong AI:** A sort of AI that can execute any task that a human can perform. It has the potential to transform many parts of our lives.

Machine Learning

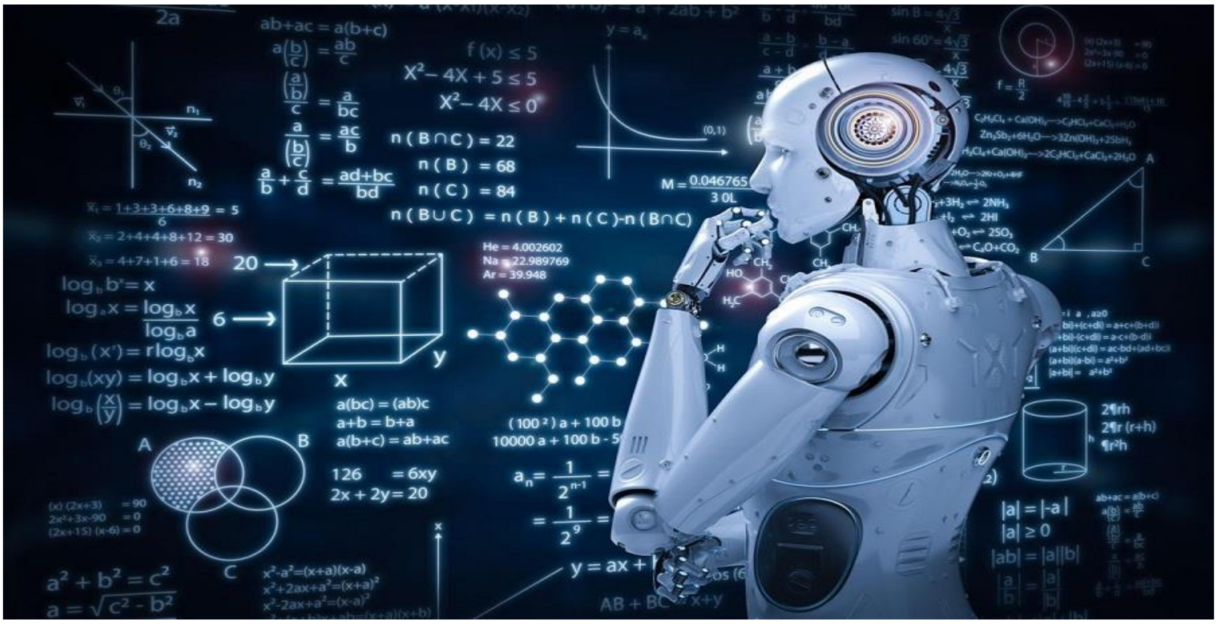


Figure 2: Presentation of Machine Learning. Source : *(javatpoint, 2023)*

Machine learning is a developing technique that allows computers to learn autonomously from historical data. Machine learning employs a variety of algorithms to construct mathematical models and make predictions based on past data or information. It is being utilized for a variety of activities including image recognition, audio recognition, email filtering, Facebook auto-tagging, recommender systems, and many others (javatpoint, 2023).

2.2.6 What is Machine Learning

In the real world, we are surrounded by humans who can learn everything from their experiences with their learning capability, and we have computers or machines which work on our instructions. But can a machine also learn from experiences or past data like a human does? So here comes the role of **Machine Learning**.

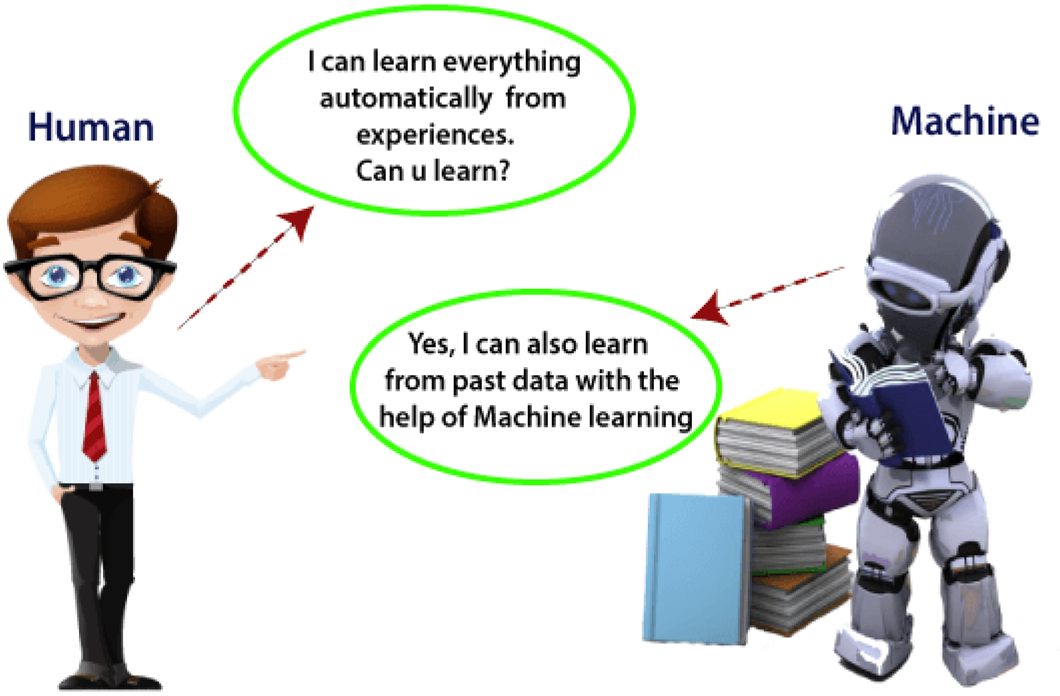

Human
I can learn everything automatically from experiences. Can u learn?
Machine
Yes, I can also learn from past data with the help of Machine learning

Figure 3 :How computer learn from the data and past experiences. Source :(TutorialsTeacher,2022)

Machine Learning is said as a subset of **artificial intelligence** that is mainly concerned with the development of algorithms which allow a computer to learn from the data and past experiences on their own. The term machine learning was first introduced by **Arthur Samuel** in **1959**. We can define it in a summarized way as:

With the help of sample historical data, which is known as **training data**, machine learning algorithms build a **mathematical model** that helps in making predictions or decisions without being explicitly programmed. Machine learning brings computer science and statistics together for creating predictive models. Machine learning constructs or uses the algorithms that learn from historical data. The more we will provide the information, the higher will be the performance (Samuel, 2023).

2.2.7 How does Machine Learning work

A Machine Learning system learns from previous data, constructs prediction models, and forecasts the output for new data. The amount of data influences the accuracy of projected output since a large amount of data helps to construct a better model that predicts the output more precisely.

If we have a complex problem that requires some predictions, instead of creating code for it, we can simply input the data to generic algorithms, and the machine will develop the logic based on the data and forecast the output. Machine learning has altered our perspective on the issue. The following block diagram describes how the Machine Learning algorithm works (javatpoint, 2023):

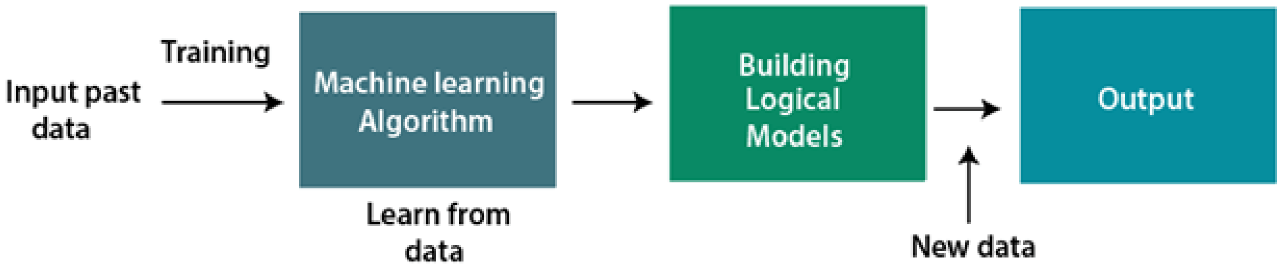


Figure 4: How does machine learning wok. Source: (TutorialsTeacher,2022)

Features of Machine Learning:

- Machine learning uses data to detect various patterns in a given dataset.
- It can learn from past data and improve automatically.
- It is a data-driven technology.

Machine learning is much similar to data mining as it also deals with the huge amount of the data.

2.2.8 Need for Machine Learning

The demand for machine learning is growing by the day. The necessity for machine learning stems from its ability to perform tasks that are too complex for a human to perform directly. As humans, we have some limits because we cannot access large amounts of data manually, therefore we need computer systems, and this is where machine learning comes in to help us.

We can train machine learning algorithms by feeding them massive amounts of data and allowing them to automatically examine the data, develop models, and predict the desired output. The performance of the machine learning algorithm is defined by the cost function and is dependent on the amount of data. We can save both time and money by utilizing machine learning.

The value of machine learning can be simply grasped by looking at its applications. Machine learning is currently employed in self-driving cars, cyber fraud detection, face recognition, and Facebook friend suggestion, among other applications. Several big companies, including Netflix and Amazon, have built machine learning models that monitor user interest and recommend products based on that data (microfocus, 2023).

2.2.9 Classification of Machine Learning

At a broad level, machine learning can be classified into three types:

1. **Supervised learning**
2. **Unsupervised learning**
3. **Reinforcement learning**

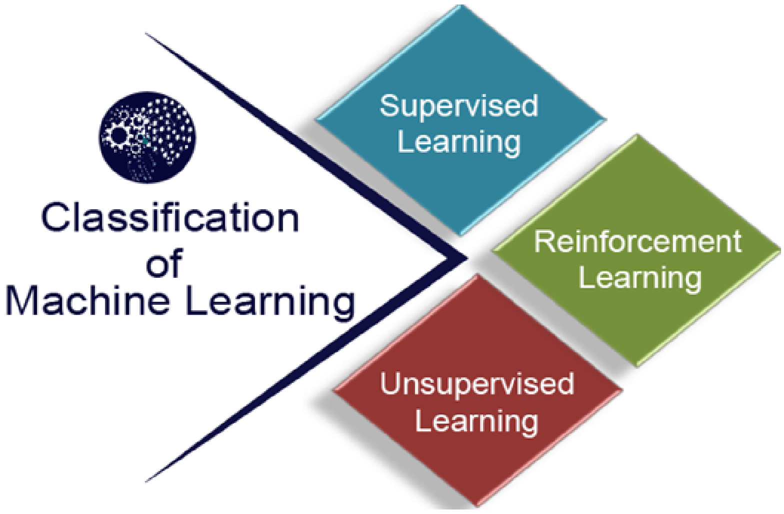


Figure 5:Classification of Machine Learning. Source (TutorialsTeacher,2023)

1) Supervised Learning

Supervised learning is a form of machine learning method in which we feed sample labeled data to the machine learning system in order to train it, and it predicts the output based on that.

The system builds a model using labeled data to interpret the datasets and learn about each data. After training and processing, we test the model by supplying sample data to see if it predicts the precise output or not.

The purpose of supervised learning is to connect input and output data. Supervised learning is dependent on monitoring, such like when a student learns something under the observation of a teacher. Spam filtering is one example of supervised learning (Munappy, Jan Bosch, Helena Holmström Olsson, & Anders Arpteg, september 2022).

Supervised learning can be grouped further in two categories of algorithms:

i. **Classification**
ii. **Regression**

2) Unsupervised Learning

Unsupervised learning is a type of learning in which a machine learns without being supervised.

The machine is trained given a set of unlabeled, classified, or categorized data, and the algorithm is expected to act on the data without supervision. Unsupervised learning attempts to restructure input data into new features or groups of objects with similar patterns.

In unsupervised learning, we don't have a predetermined result. The machine tries to find useful insights from the huge amount of data. It can be further classifieds into two categories of algorithms (Brownlee J. , 2020):

i. **Clustering**
ii. **Association**

3) Reinforcement Learning

Reinforcement learning is a feedback-based learning strategy in which a learning agent is rewarded for correct actions and penalized for incorrect actions. With these feedbacks, the agent automatically learns and improves its performance. The agent interacts with and investigates the environment in reinforcement learning. An agent's purpose is to earn the most reward points possible, so it enhances its performance.

Reinforcement learning is demonstrated by the robotic dog, which automatically learns the movement of his arms.

2.2.10 Prerequisites

Before learning machine learning, you must have the basic knowledge of followings so that you can easily understand the concepts of machine learning:

i. Fundamental knowledge of probability and linear algebra.
ii. The ability to code in any computer language, especially in Python language.
iii. Knowledge of Calculus, especially derivatives of single variable and multivariate functions.

2.2.11 Deep learning vs. Machine learning vs. Artificial Intelligence

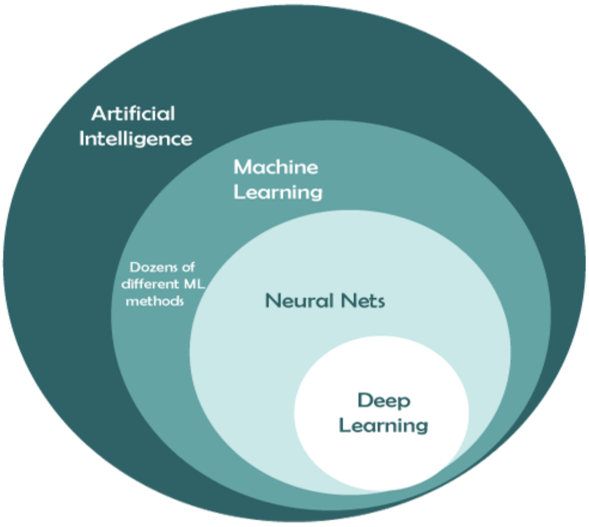


Figure 6: Deep learning vs. Machine learning vs. Artificial Intelligence. Source : *(javatpoint, 2023)*

Deep Learning, Machine Learning, and Artificial Intelligence are the most used terms on the internet for IT folks.

However, all three technologies are linked to one another. Artificial intelligence (AI) is a broad term that encompasses both machine learning and deep learning. Deep learning and machine learning are both subsets of artificial intelligence.

Because these technologies appear similar, many people believe that 'Deep Learning, Machine Learning, and Artificial Intelligence' are all interchangeable. However, while all of these technologies are utilized to create intelligent machines or apps that behave like humans, they differ in terms of functionality and scope.

It means that while these three names are frequently used interchangeably, they do not always refer to the same thing. Let us first define the key distinction between deep learning and machine learning (javatpoint, 2023).

2.2.12 Python Programming

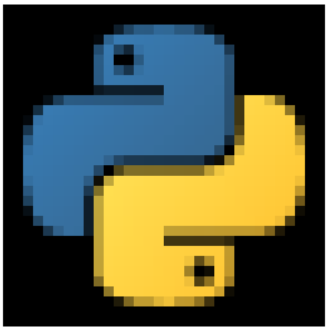


Figure 7:Python Programming Icone. Source : *(javatpoint, 2023)*

Python is a general-purpose high-level programming language. It is an open source language, released under a GPL-compatible license. Python is used in various types of applications such as data science, Machine Learning, Web Development, Image Processing, Game Development, Embedded Systems and IoT, etc.

Guido Van Rossum conceived Python in the late 1980s. It was released in 1991 at Centrum Wiskunde & Informatica (CWI) in the Netherlands as a successor to the ABC language. He named this language after a popular comedy show called 'Monty Python's Flying Circus' (and not after Python-the snake) (tutorialsteacher, 2023)

In the last few years, its popularity has increased immensely. According to stackoverflow.com's recent survey, Python is in the top three (stackoverflow, 2023)

Python Features:

- Python is an interpreter-based language, which allows the execution of one instruction at a time.
- Extensive basic data types are supported e.g., numbers (floating point, complex, and unlimited-length long integers), strings (both ASCII and Unicode), lists, and dictionaries.
- Variables can be strongly typed as well as dynamic typed.
- Supports object-oriented programming concepts such as class, inheritance, objects, module, namespace etc.
- Cleaner exception handling support.
- Supports automatic memory management.

- Various built-in and third-party modules, which can be imported and used independently in the Python application

Python Advantages

- Python provides enhanced readability. For that purpose, uniform indents are used to delimit blocks of statements instead of curly brackets, like in many languages such as C, C++, and Java.
- Python is free and distributed as open-source software. A large programming community is actively involved in the development and support of Python libraries for various applications such as web frameworks, mathematical computing, and data science.
- Python is a cross-platform language. It works equally on different OS platforms like Windows, Linux, Mac OSX, etc. Hence Python applications can be easily ported across OS platforms (stackoverflow, 2023).
- Python supports multiple programming paradigms including imperative, procedural, object-oriented, and functional programming styles.
- Python is an extensible language. Additional functionality (other than what is provided in the core language) can be made available through modules and packages written in other languages (C, C++, Java, etc.) (stackoverflow, 2023).
- A standard DB-API for database connectivity has been defined in Python. It can be enabled using any data source (Oracle, MySQL, SQLite etc.) as a backend to the Python program for storage, retrieval, and processing of data.
- The standard distribution of Python contains the Tkinter GUI toolkit, which is the implementation of a popular GUI library called Tcl/Tk. An attractive GUI can be constructed using Tkinter. Many other GUI libraries like Qt, GTK, WxWidgets, etc. are also ported to Python (stackoverflow, 2023).
- Python can be integrated with other popular programming technologies like C, C++, Java, ActiveX, and CORBA

Python Tools and Frameworks

The following lists important tools and frameworks to develop different types of Python applications (Khatri, 2023):

- **Web Development :** Django, Pyramid, Bottle, Tornado, Flask, web2py
- **GUI Development** : tkInter, PyGObject, PyQt, PySide, Kivy, wxPython
- **Scientific and Numeric** : SciPy, Pandas, IPython

- **Software Development** : Buildbot, Trac, Roundup
- **System Administration** : Ansible, Salt, OpenStack

2.2.13 What is Flask Python

Figure 8:Flask web framework. Source : *(javatpoint, 2023)*

Flask is a web framework, it’s a Python module that lets you develop web applications easily. It’s has a small and easy-to-extend core: it’s a microframework that doesn’t include an ORM (Object Relational Manager) or such features.

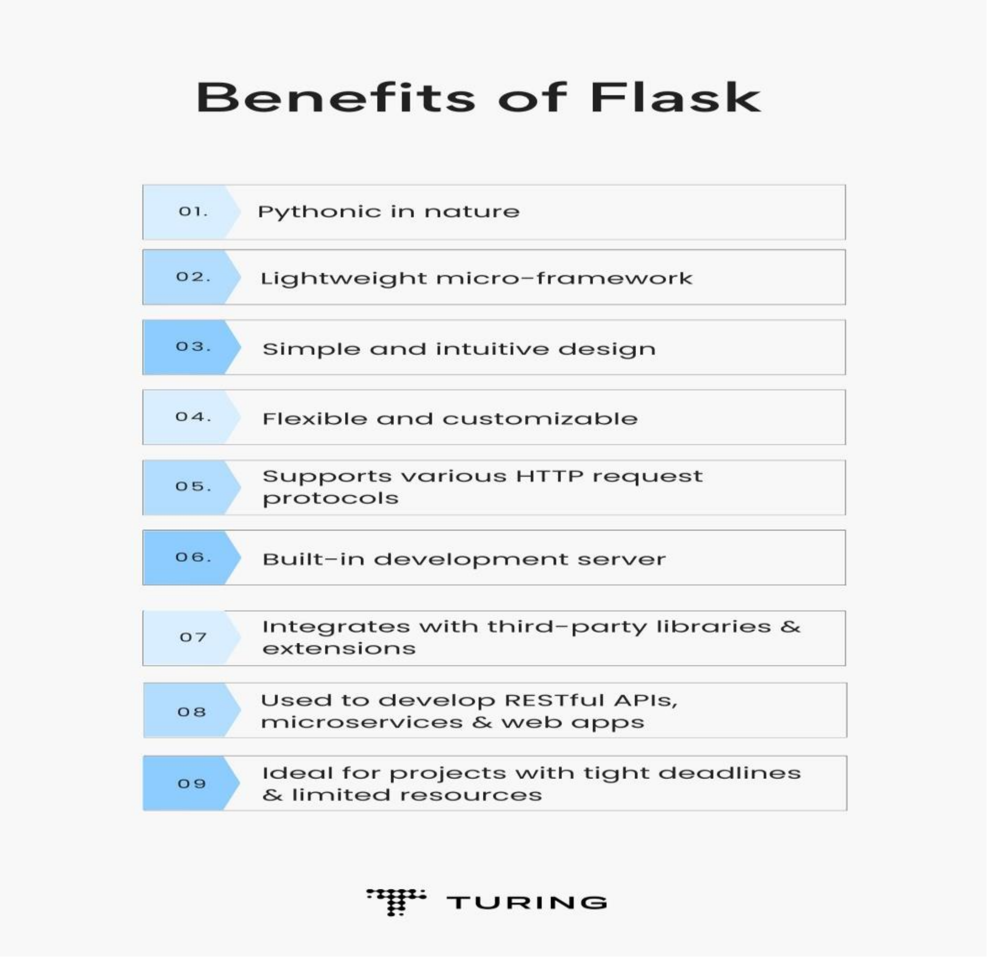
It does have many cool features like url routing, template engine. It is a WSGI web app framework. (pythonbasics, 2023)

2.2.14 What is a Web Framework?

A Web Application Framework or a simply a Web Framework represents a collection of libraries and modules that enable web application developers to write applications without worrying about low-level details such as protocol, thread management, and so on (turing, 2023).

2.2.15 What is Flask?

Flask is a web application framework written in Python. It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco. Flask is based on the Werkzeg WSGI toolkit and the Jinja2 template engine.Both are Pocco projects (tutoriel, 2023).



Benefits of Flask

01. Pythonic in nature

02. Lightweight micro-framework

03. Simple and intuitive design

04. Flexible and customizable

05. Supports various HTTP request protocols

06. Built-in development server

07 Integrates with third-party libraries & extensions

08 Used to develop RESTful APIs, microservices & web apps

09 Ideal for projects with tight deadlines & limited resources

Figure 9: Benefits of Flask. Source : (turing, 2023)*:*

2.2.16 Why deploy machine learning models?

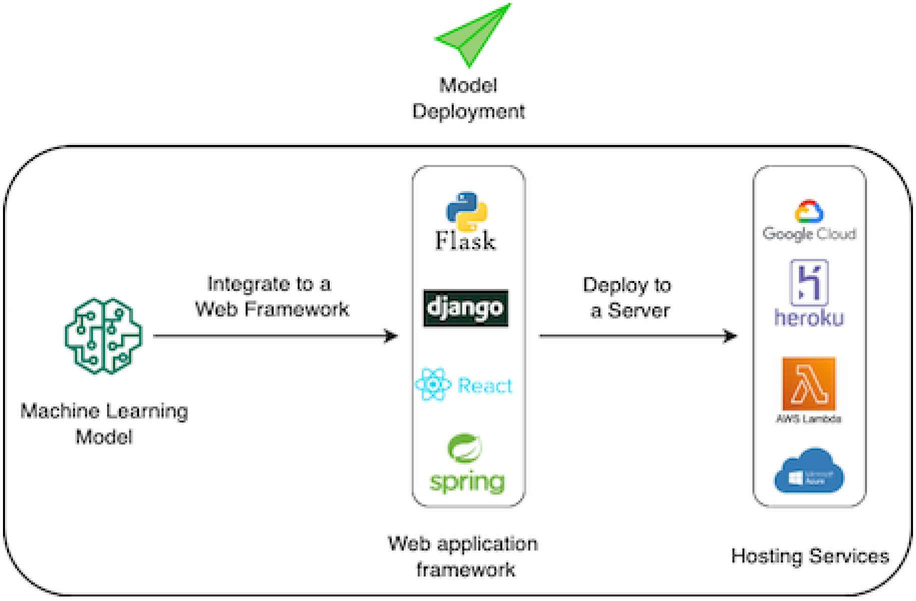


Figure 10:Deploy machine laerning modeles on Web Framework. Source: (turing, 2023)

Assume you've built a machine learning model to count the number of cars traveling through a specific road using a camera mounted on it. An ML engineer creates the model locally at first. Once fully designed and tested, it must be moved to production or it will be useless in real life. As a result, machine learning models must be deployed in production. Before considering your model for deployment, the following elements are necessary (turing, 2023):

1. Portability: The concept of "portability" in software development refers to the ability of the software to be easily transferred from one system or machine to another, with low response time and minimal effort required for rewriting (turing, 2023):.

2. Scalability: "Scalability" refers to the ability of a software model to handle larger workloads without a drop in performance without the need for major redesign or reconfiguration. A scalable model is designed to maintain its performance even when handling large amounts of data or complex tasks (turing, 2023):.

2.2.17 Steps to deploy a machine learning model with Flask and Docker

We will use Docker as a container to deploy the Flask app. A container is similar to a virtual machine except that it does not have its own resources. Rather, it shares them with the host machine. This helps to deploy the app easily by installing all the dependencies (turing, 2023).

A self-contained and efficient software package called a Docker container image encompasses all the necessary components to execute an application, including code, runtime, system tools, system libraries, and configurations

1. **Create a machine learning model :** In this step, we will create a ML model that will be used for deployment.
2. **Create a REST API with Flask:** Now, we will create a Flask API that will be called later for inferencing.
3. **Create a Docker Image :** After creating the service, the initial step will involve defining the Docker image, which is critical for the process. Defining the Dockerfile contents is essential to include all the necessary dependencies and copy the application's content into the container.
4. **Run Docker :** Here, we will build a Docker container for the service. Note that it is necessary to install Docker before this step is executed.

2.2.18 DOCKER

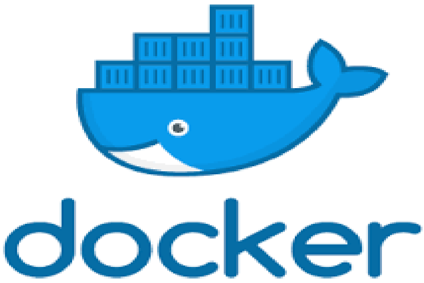


Figure 11:Docker Icone. Source : (Naik, 2017).

Docker is a tool that can encapsulate an application and its dependencies in a secure container that may be run on any server. It is not virtualization, but containerization, a more lightweight variant that relies on specific parts of the host machine to function.

Docker's container technology can be used to extend distributed systems so that they can run autonomously from a single physical computer or a single instance per node. This enables the nuds to be deployed as and when resources become available (Naik, 2017).

2.2.19 KUBERNETES

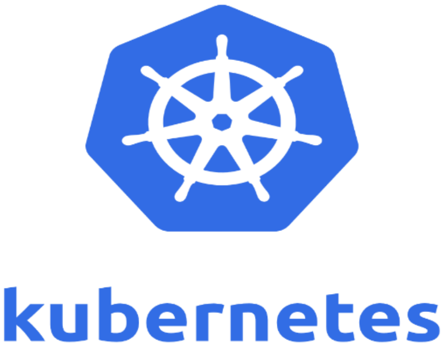


Figure 12:Kubernetes icone. Source : (kubernetes, 2023)

Kubernetes (also known as "K8s2") is an open source system that aims to provide a "platform for automating the deployment, installation, and operation of application containers on server clusters." It is compatible with a wide range of container technologies and is frequently used in conjunction with Docker. It was originally designed by Google and then offered to the Cloud Native Computing Foundation. Kubernetes allows us to avoid load balancing by starting multiple instances as needed and ensuring the service load-balancing, which determines which container to direct an incoming request to given that there are multiple instances of the application

running (kubernetes, 2023).

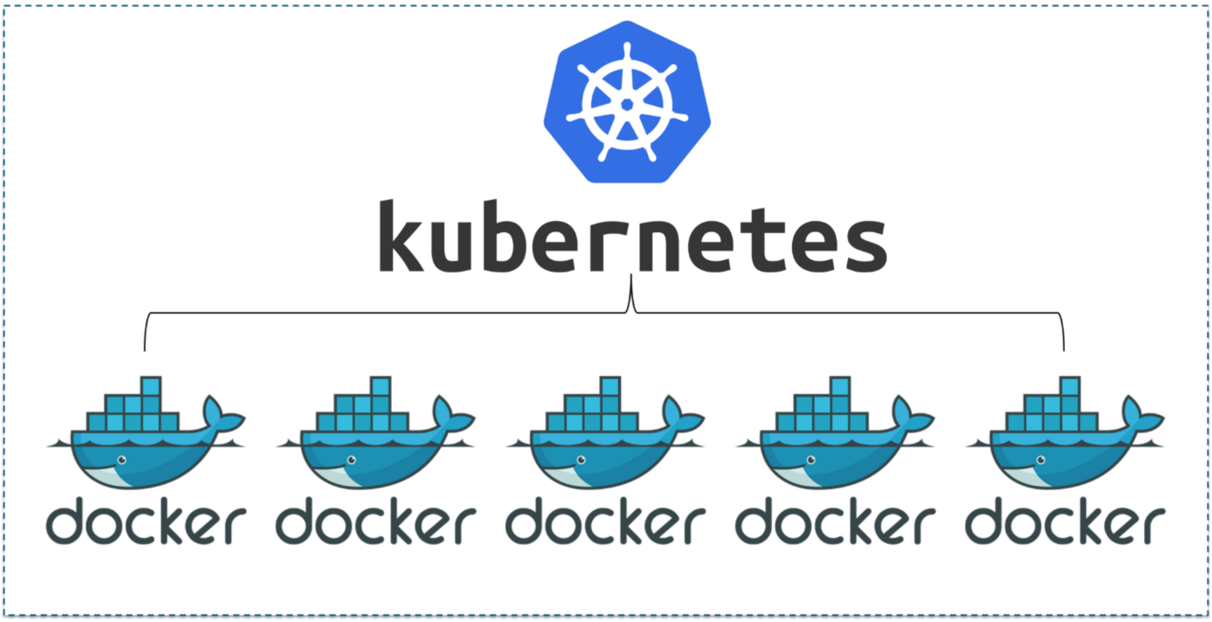


Figure 13:multiple instances of the application running on Kubernetes. Source : *(kubernetes, 2023)*

2.2.19 What is UML

Unified Modeling Language (UML) is a general-purpose modelling language. The main aim of UML is to define a standard way to **visualize** the way a system has been designed. It is quite similar to blueprints used in other fields of engineering (.visual-paradigm, 2023).

UML is **not a programming language**, it is rather a visual language. We use UML diagrams to portray the **behavior and structure** of a system. UML helps software engineers, businessmen and system architects with modelling, design and analysis. The Object Management Group (OMG) adopted Unified Modelling Language as a standard in 1997. Its been managed by OMG ever since. International Organization for Standardization (ISO) published UML as an approved standard in 2005. UML has been revised over the years and is reviewed periodically (visual-paradigm, 2023).

Do we really need UML?

- Complex applications need collaboration and planning from multiple teams and hence require a clear and concise way to communicate amongst them.
- Businessmen do not understand code. So UML becomes essential to communicate with non programmers essential requirements, functionalities and processes of the system.
- A lot of time is saved down the line when teams are able to visualize processes, user interactions and static structure of the system.

UML is linked with **object oriented** design and analysis. UML makes the use of elements and forms associations between them to form diagrams. Diagrams in UML can be broadly classified as (artoftesting, 2023):

1. **Structural Diagrams –** Capture static aspects or structure of a system. Structural Diagrams include: Component Diagrams, Object Diagrams, Class Diagrams and Deployment Diagrams.
2. **Behavior Diagrams –** Capture dynamic aspects or behavior of the system. Behavior diagrams include: Use Case Diagrams, State Diagrams, Activity Diagrams and Interaction Diagrams.

The image below shows the hierarchy of diagrams according to UML 2.2

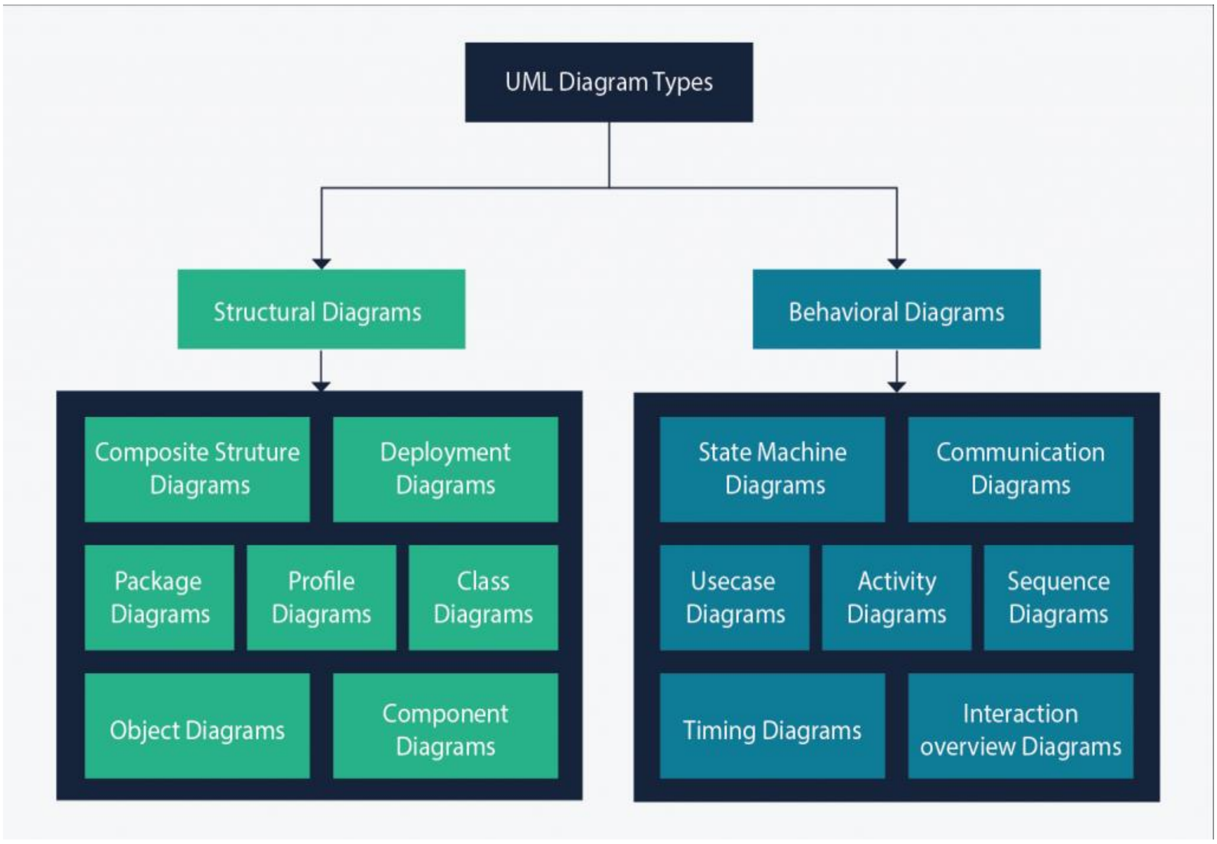


Figure 14:the hierarchy of diagrams according to UML 2.2. Source : *(.visual-paradigm, 2023)*

Object Oriented Concepts Used in UML –

1. **Class** – A class defines the blue print i.e. structure and functions of an object.
2. **Objects** – Objects help us to decompose large systems and help us to modularize our system. Modularity helps to divide our system into understandable components so that we can build our system piece by piece. An object is the fundamental unit (building block) of a system which is used to depict an entity.
3. **Inheritance** – Inheritance is a mechanism by which child classes inherit the properties of their parent classes.
4. **Abstraction** – Mechanism by which implementation details are hidden from user.
5. **Encapsulation** – Binding data together and protecting it from the outer world is referred to as encapsulation.
6. **Polymorphism** – Mechanism by which functions or entities are able to exist in different forms.

Additions in UML 2.0 –

- Software development methodologies like agile have been incorporated and scope of original UML specification has been broadened.
- Originally UML specified 9 diagrams. UML 2.0 has increased the number of diagrams from 9 to 13. The four diagrams that were added are: timing diagram, communication diagram, interaction overview diagram and composite structure diagram. UML 2.0 renamed statechart diagrams to state machine diagrams.
- UML 2.0 added the ability to decompose software system into components and sub-components.

Structural UML Diagrams

1. **Class Diagram –** The most widely use UML diagram is the class diagram. It is the building block of all object oriented software systems. We use class diagrams to depict the static structure of a system by showing system's classes,their methods and attributes. Class diagrams also help us identify relationship between different classes or objects.
2. **Composite Structure Diagram –** We use composite structure diagrams to represent the internal structure of a class and its interaction points with other parts of the system. A composite structure diagram represents relationship between parts and their configuration which determine how the classifier (class, a component, or a deployment node) behaves. They represent internal structure of a structured classifier making the use of parts, ports, and connectors. We can also model collaborations using composite structure diagrams. They are similar to class diagrams except they represent individual parts in detail as compared to the entire class.
3. **Object Diagram –** An Object Diagram can be referred to as a screenshot of the instances in a system and the relationship that exists between them. Since object diagrams depict behaviour when objects have been instantiated, we are able to study the behaviour of the system at a particular instant. An object diagram is similar to a class diagram except it shows the instances of classes in the system. We depict actual classifiers and their relationships making the use of class diagrams. On the other hand, an Object Diagram represents specific instances of classes and relationships between them at a point of time.
4. **Component Diagram –** Component diagrams are used to represent the how the physical components in a system have been organized. We use them for modelling implementation details. Component Diagrams depict the structural relationship

between software system elements and help us in understanding if functional requirements have been covered by planned development. Component Diagrams become essential to use when we design and build complex systems. Interfaces are used by components of the system to communicate with each other.

5. **Deployment Diagram –** Deployment Diagrams are used to represent system hardware and its software.It tells us what hardware components exist and what software components run on them.We illustrate system architecture as distribution of software artifacts over distributed targets. An artifact is the information that is generated by system software. They are primarily used when a software is being used, distributed or deployed over multiple machines with different configurations.
6. **Package Diagram –** We use Package Diagrams to depict how packages and their elements have been organized. A package diagram simply shows us the dependencies between different packages and internal composition of packages. Packages help us to organise UML diagrams into meaningful groups and make the diagram easy to understand. They are primarily used to organise class and use case diagrams.

Behavior Diagrams

1. **State Machine Diagrams –** A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It’s a behavioral diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as **State machines** and **State-chart Diagrams** . These terms are often used interchangeably.So simply, a state diagram is used to model the dynamic behavior of a class in response to time and changing external stimuli.
2. **Activity Diagrams –** We use Activity Diagrams to illustrate the flow of control in a system. We can also use an activity diagram to refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagrams. So, we basically depict workflows visually using an activity diagram.An activity diagram focuses on condition of flow and the sequence in which it happens. We describe or depict what causes a particular event using an activity diagram(visual-paradigm, 2023).
3. **Use Case Diagrams –** Use Case Diagrams are used to depict the functionality of a system or a part of a system. They are widely used to illustrate the functional requirements of the system and its interaction with external agents(actors). A use case is basically a diagram representing different scenarios where the system can be used.

A use case diagram gives us a high level view of what the system or a part of the system does without going into implementation details.

4. **Sequence Diagram –** A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place.We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems (visual-paradigm, 2023).
5. **Communication Diagram –** A Communication Diagram(known as Collaboration Diagram in UML 1.x) is used to show sequenced messages exchanged between objects. A communication diagram focuses primarily on objects and their relationships. We can represent similar information using Sequence diagrams,however, communication diagrams represent objects and links in a free form.
6. **Timing Diagram –** Timing Diagram are a special form of Sequence diagrams which are used to depict the behavior of objects over a time frame. We use them to show time and duration constraints which govern changes in states and behavior of objects(visual-paradigm, 2023)..

Interaction Overview Diagram – An Interaction Overview Diagram models a sequence of actions and helps us simplify complex interactions into simpler occurrences. It is a mixture of activity and sequence diagrams.

Chapter III: RESEARCH METHODOLOGY

3.1 Overview

For conducting this research activity, we have used different approaches. Thus, this section gives a brief explanation about research methodology approaches used by the researcher throughout in this study.

These include Machine Learning models Training and Evaluation (Data Collection, Data Visualization, Data Preparation and Wrangling), Project Management Tools (Gannt and PERT) and Agile development Approach using UML.

3.2 Project management tools (PERT & GANTT CHART)

PERT charts are best used during the planning phase of a project, and allow users to map out the full scope. Gantt charts are more helpful once a project is underway, as they can be adjusted if the scope changes. PERT charts have a specific formula used to calculate time estimates for tasks.

3.2.1 Provisional planning of the project

To track the progress of our project, we divided it into 14 tasks with a total execution time of 80 days.

The task identification table and their anteriority relations are shown below:

Table 1:Table 1: List of Tasks

Tasks	Designation	Duration(Day)	The anterior Task
A	Contact with the provincial health division authorities	1	--
B	Data collection	5	A
C	Analyzing and purifying the collected data database	7	B
D	Training from data	1	C
E	Building logical model	1	D
F	Application coding	12	E
H	Code Documentation	1	F
H	Testing the final deployment	3	G
I	Final bug fixes	3	G,H
L	Official launch of the application	1	I
TOTAL		35 Days	

Based on the daily unit expenses of carrying out these operations, we were able to estimate the entire cost of the project. The outcome of this strategy is shown in the table below.

Table 2:List of Task Based on the daily unit expenses

Tasks	Designation	Duration	The anterior Task	unit price	Total amount
A	Contact with the provincial health division authorities	1	--	10$	10$
B	Data collection	5	A	100$	2000$
C	Analyzing and purifying the collected data database	7	B	40$	280$
D	Training from data	1	C	100$	400$
E	Building logical model	1	D	400$	400$
F	Application coding	12	E	50$	600$
H	Code Documentation	1	F	50$	50$
H	Testing the final deployment	3	G	20$	60$
I	Final bug fixes	3	G,H	20$	60$
L	Official launch of the application and training of users	5	I	100$	500$
TOTAL		35 Days			4360$

Following this critical step, we will first depict these tasks on a PERT chart, followed by a Gantt chart. These diagrams allowed us to pinpoint precisely when the execution of a task should begin based on the precedence restrictions between tasks and when it should have already been completed in order to meet the task's deadline. The project's execution.

This is how we will identify the critical path in the project's task execution. It is made up of a series of jobs whose earliest and latest start dates are the same. It is critical to remember that a delay at the start of one of these jobs directly implies a delay in the entire project. In our scenario, the essential job path will be represented on a PERT graph by a succession of green nodes and by red lines on the Gantt chart.

3.3 Data mining Approach

Data mining is looking for patterns in huge data stores. This process brings useful ways, and thus we can make conclusions about the data. This also generates new information about the data which we possess already. The methods include tracking patterns, classification, association, outlier detection, clustering, regression, and prediction. It is easy to recognize patterns, as there can be a sudden change in the data given. We have collected and categorized the data based on different sections to be analyzed with the categories. Clustering groups the data based on the similarities of the data (Pedamkar, 2023).

3.3.1 Data Collection and Visualization

The researcher gathered several forms of data from two government entities in order to carry out this research study. These data contain ebola cases and the history of population growth in the provinces of North Kivu and Ituri. The next paragraphs describe in detail how these data were gathered.

Ebola Data

The Ebola dataset includes historical detections, the number of people confirmed to be infected with Ebola by hospitals and clinics in Noth Kivu and Ituri Provinces, and is aggregated weekly.

These are the records of NORTH KIVU's provincial health division (DPS) for the two-year period from 2018 to 2019, as shown in Figure 3.

As shown in Figure 3, which estimates the number of ebola cases in Noth Kivu and Ituri Provinces between 2018 and 2019, the number of confirmed ebola cases climbed steadily in 2019.

3.3.2 Population Data

Data Mining Techniques

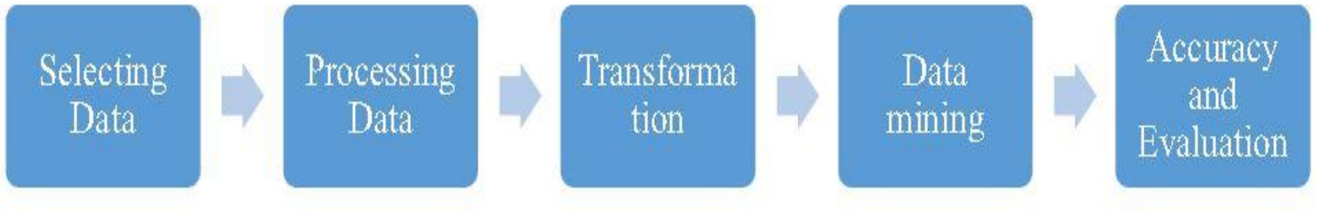


Figure 15:Schematic diagram representing the data mining process. Source : *(javatpoint, 2023)*

3.3 Predictive Models Modelling

This section provides a full description and implementation processes for the various Machine Learning Classification algorithms employed by the researcher during the course of this research. These methods were used to map the relationship between dependent and independent variables (model input and model target). Logistic Regression, Random Forest Classifier, K-Nearest Neighbors Classifier, Multi-Layer Perceptron Classifier, Decision Tree Classifier, and Gradient Boosting Classifier are among them. These algorithms were chosen because they are widely used in solving machine learning classification issues and have a considerable amount of documentation. The next paragraphs provide a brief description of each of the algorithms mentioned above.

3.3.1 Logistic Regression

A Logistic Regression is a sort of supervised machine learning method used to create predictions whether the target variables are discrete or categorical. It is widely used to solve binary classification issues such as spam identification, cancer diagnosis, and anomaly detection. Unlike Linear Regression, which predicts unbound values, Logistic Regression predicts a range of values (Stojiljković, 2022).

F(X): Logistic Regression is the mathematical expression.

F(X) =sigmoid (WX+ b)

Equation 1: Logistic Regression is the mathematical expression

Here, **X** is the input feature vector. **W**, **b** and **sigmoid** are the weight vector, bias and activation function respectively. Weight vector and bias are the model parameters to be identified during of model training. The sigmoid function or activation function is used for mapping the values between 1 and 0. If the output of the sigmoid function is above 0.5 we can classify this as 1 and 0 is the output is below 0.5 (Navlani, 2022).

sigmoid (x) = 1 / (1 + e(- x))

Equation 2: representation of the sigmoid function

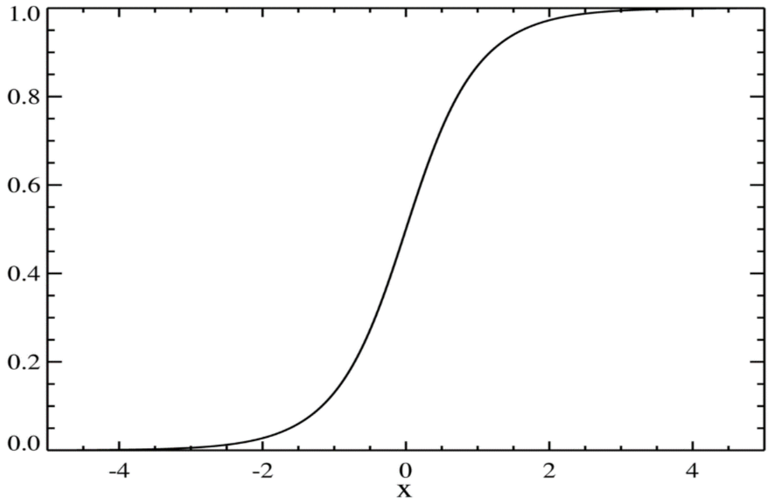
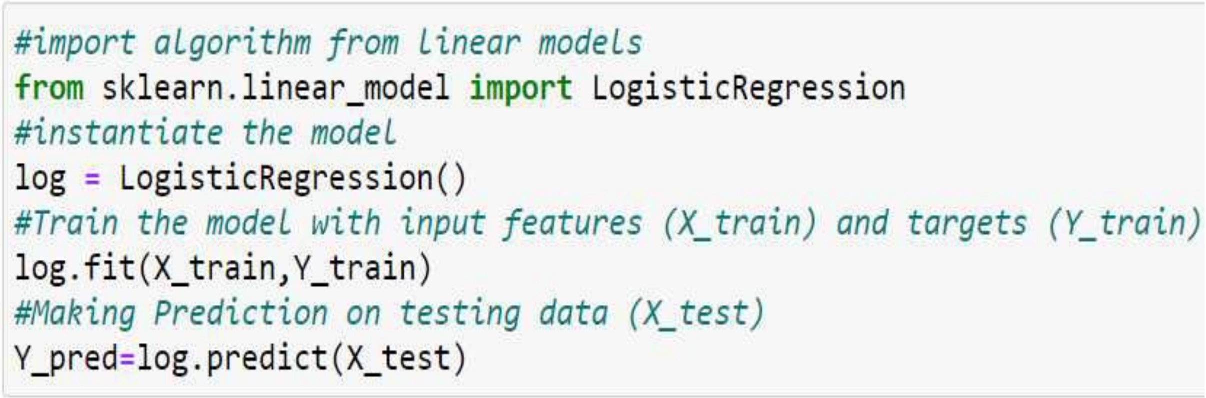


Figure 16: Sigmoid function. Source : *(Navlani, 2022)*

Python Implementation:



```
#import algorithm from linear models
from sklearn.linear_model import LogisticRegression
#instantiate the model
log = LogisticRegression()
#Train the model with input features (X_train) and targets (Y_train)
log.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=log.predict(X_test)
```

Figure 17: Python Implementation of Sigmoid function. Source: (stackoverflow,2022)

3.3.2 Décision Tree Classifier

Decision tree Classifier is a supervised machine learning algorithm used in machine learning and in statistics when the target variables are categorical. This predicting modelling approach uses a tree-like graph as a predictive model where observations are represented the branches and target values or the actual output or class represented in the leaves. The goal of this algorithm is to build a predictive model that can predicts the target value by learning decision rules identified from the features.

These rules are implemented by using if-then-else statements (Shubham, 2022). Decision trees generates predictions by sorting them down the tree from the root to some leaf node, with the leaf node providing the classification to the observations.

Let take an example of problem to determine

if someone can go to swim based on the weather conditions.

The figure 20 generates different answers (predictions) based on different climates factors.

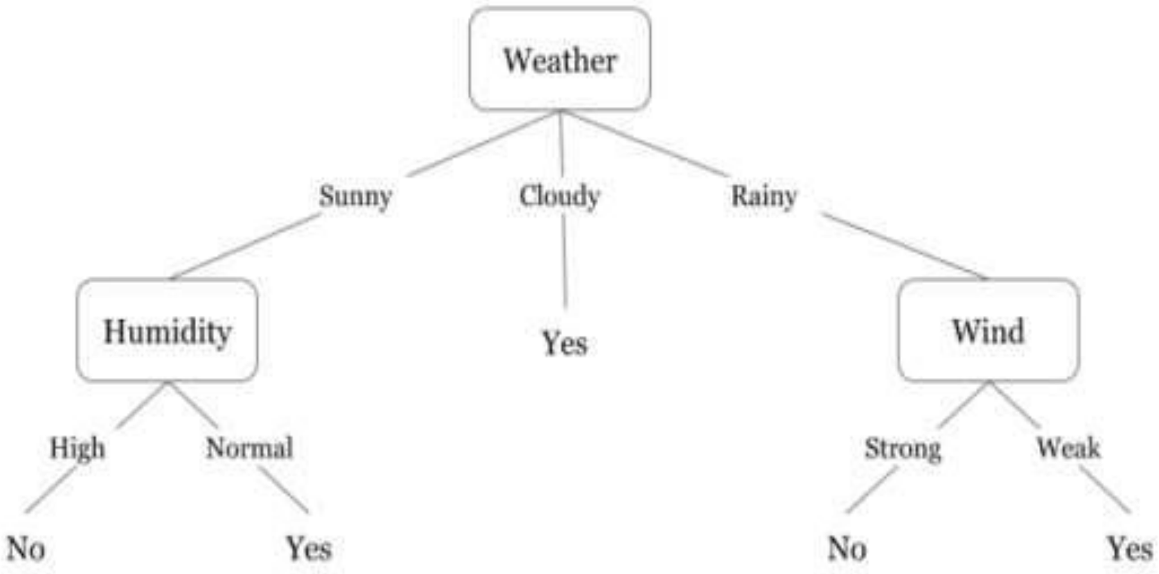
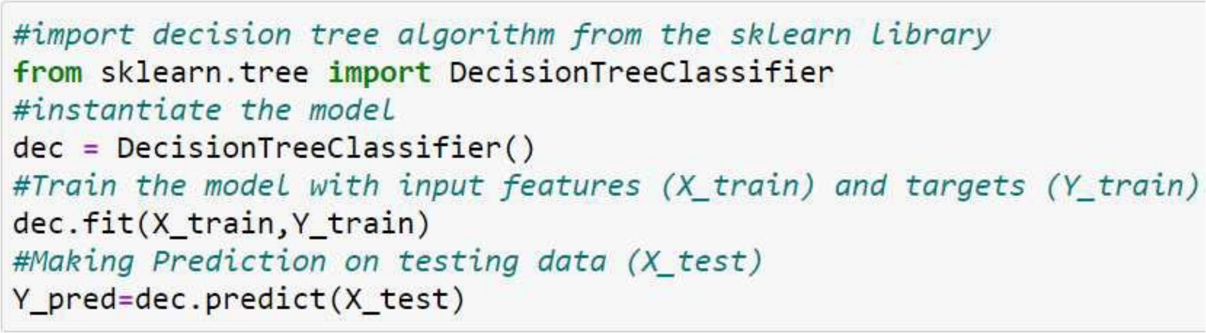


Figure 18 : A decision tree for play concept. Source : *(javatpoint, 2023)*

The possible answers for swimming concept given in above tree are represented in the table 4

Python Implementation:



```
#import decision tree algorithm from the sklearn library
from sklearn.tree import DecisionTreeClassifier
#instantiate the model
dec = DecisionTreeClassifier()
#Train the model with input features (X_train) and targets (Y_train)
dec.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=dec.predict(X_test)
```

Figure 19:Python implementation of A decision tree for play concept. Source: (stackoverflow,2022)

3.3.3 Random Forest Classifier

Decision tree machine learning can suffer from excessive variation at times, which might have a negative impact on their findings when applied to specific training data. This variance can be decreased by creating numerous predictive models in parallel from multiple samples of your training data, but these trees may be highly correlated, resulting in similar predictions. The Random Forest algorithm is a supervised machine learning algorithm that employs several trees defined from training data samples and forces them to be distinct by limiting the attributes that each model can evaluate for each sample. The class that appears frequently in the output of the many trees employed for the given training data is the final prediction (Lin, Z. Wu, L. Lin, A. Wen, & J. Li, 2017) (Cheng, P. P. K. Chan, & Z. W. Qiu, 2012).

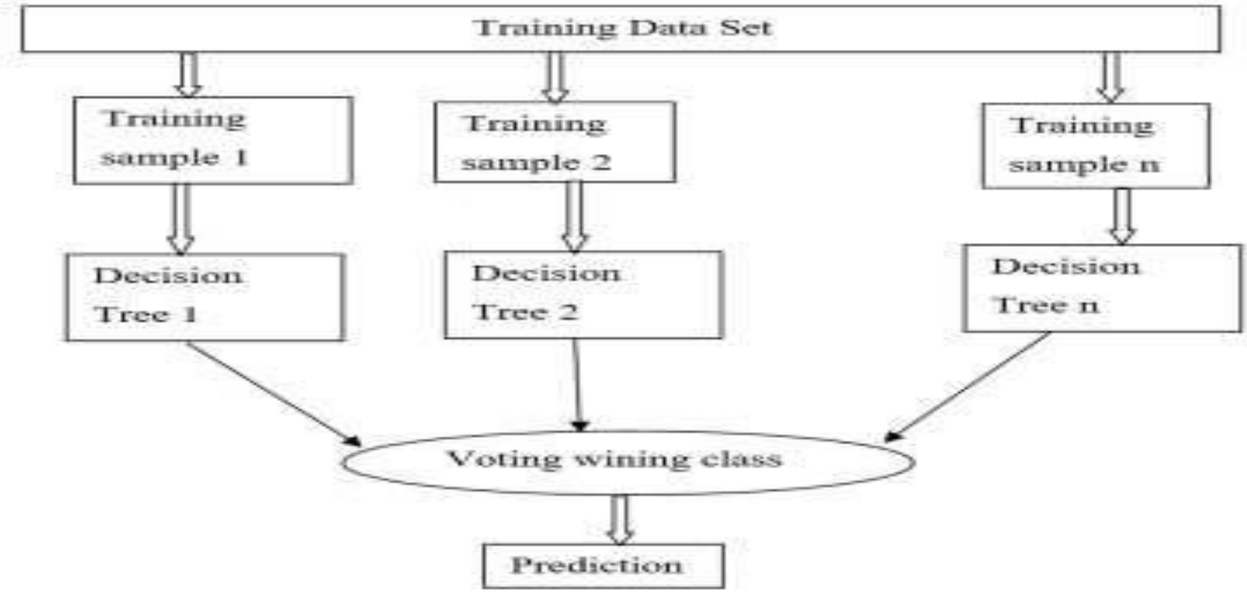
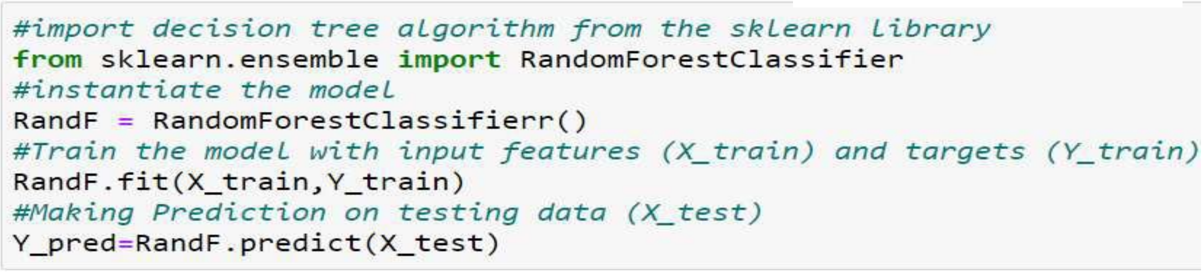


Figure 20:Building Random Forest Algorithm. Source: *(Lin, Z. Wu, L. Lin, A. Wen, & J. Li, 2017)*

Python Implementation: Source: stackoverflow.com



```
#import decision tree algorithm from the sklearn library
from sklearn.ensemble import RandomForestClassifier
#instantiate the model
RandF = RandomForestClassifierr()
#Train the model with input features (X_train) and targets (Y_train)
RandF.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=RandF.predict(X_test)
```

Figure 21: Python Implementation of Random Forest Algorithm. Source: (stackoverflow,2022)

3.4.4 Gradient Boosting Classifier

One form of ensemble technique used in machine learning to improve prediction accuracy is the gradient boosting classifier. It entails combining weak models to create a powerful prediction model. Typically, decision tree algorithms are utilized to construct a gradient boosting classifier.

When the target variables are categorical, a gradient boosting classifier is employed to create a prediction (Nelson, 2023) (KURAMA, 2023).

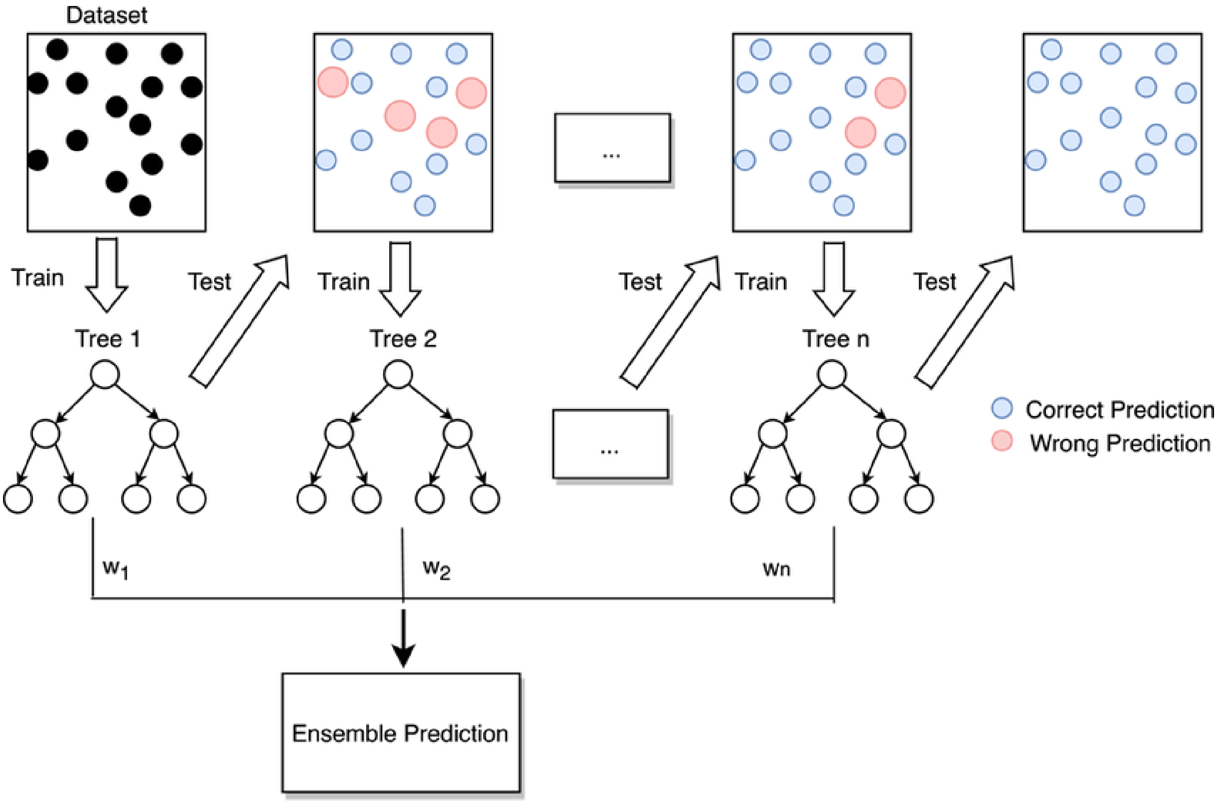
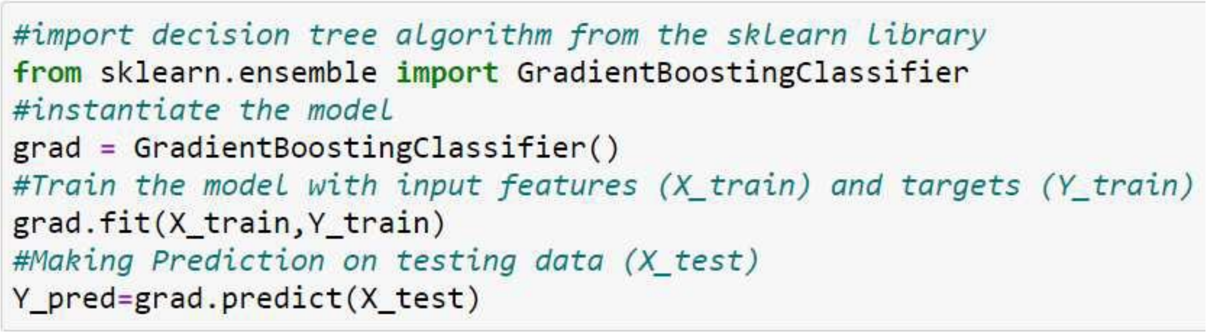


Figure 22:Building of Gradient Boosting Classifier algorithm. Source : *(KURAMA, 2023).*

Python Implementation:



```
#import decision tree algorithm from the sklearn library
from sklearn.ensemble import GradientBoostingClassifier
#instantiate the model
grad = GradientBoostingClassifier()
#Train the model with input features (X_train) and targets (Y_train)
grad.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=grad.predict(X_test)
```

Figure 23: Python Implementation of Gradient Boosting Classifier. Source: (stackoverflow,2022)

3.4.5 K-Nearest Neighbours Classifier (KNN)

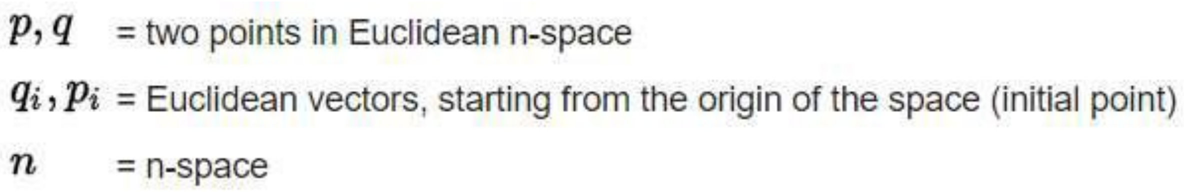
The K-Nearest Neighbours is machine learning algorithm used in finding similarities between data. During the model training phase all of the data are used for learning the similarities between data. Then during of model prediction for unseen data, the model searches through the entire dataset the K-most similar training examples to new example and the data with K-most similar instance is returned as the prediction. The algorithm states that if you are similar to your neighbours, that means that you are one of them (Brownlee J. , 2023).

In K-Nearest Neighbours, K means the number of neighbor points which contribute in voting. In KNN the voting points are selected by using Euclidean distance between the new point and the existing points and then the points with least distances are selected. The general formula of Euclidean distance is given by the following mathematical expression (Robinson, 2023)

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

Equation 3:. The general formula of Euclidean distance.Source: *(Robinson, 2023)*

Where,



p, q = two points in Euclidean n-space

q_i, p_i = Euclidean vectors, starting from the origin of the space (initial point)

n = n-space

For instance the figure 15 shows a distribution of data points with two class one in blue and other in yellow colour. Three neighbour points(K=3) is used for voting the class of the new data based on the similarities and the class with yellow is returned as the predicted class because it has more neighbours than the other class.

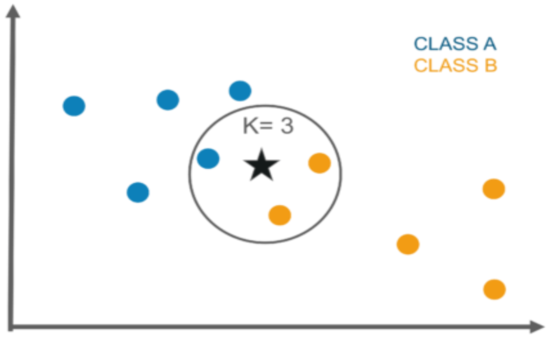
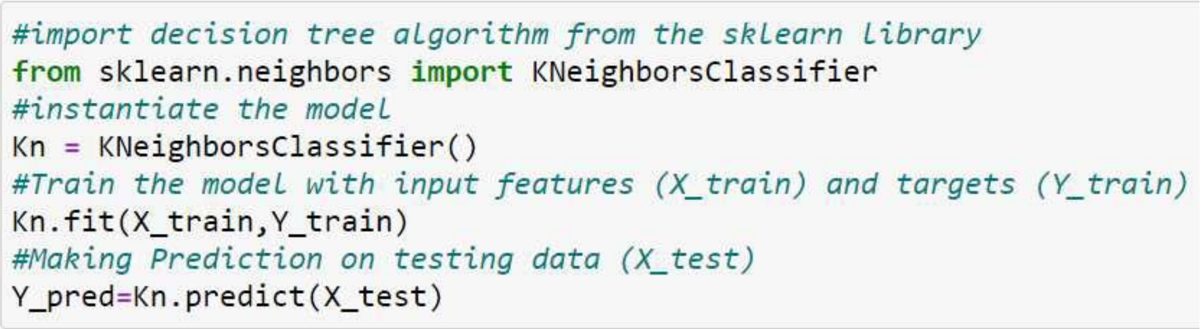


Figure 24: Graphical implementation of KNN. Source : *(Robinson, 2023)*

Python Implementation:



```
#import decision tree algorithm from the sklearn library
from sklearn.neighbors import KNeighborsClassifier
#instantiate the model
Kn = KNeighborsClassifier()
#Train the model with input features (X_train) and targets (Y_train)
Kn.fit(X_train,Y_train)
#Making Prediction on testing data (X_test)
Y_pred=Kn.predict(X_test)
```

Figure 25:Python Implementation of Graphical implementation of KNN. Source: (Stackoverflow,2022)

3.4 Agile development method using UML

This approach assists customers in refining or agreeing on a system design while ensuring everyone is satisfied with the ultimate outcome. Using unified modeling language (UML) diagrams to extend this visual approach to the agile development starting point might help deliver real outcomes for a project (Fenn, 2023)

a. Use case Diagram

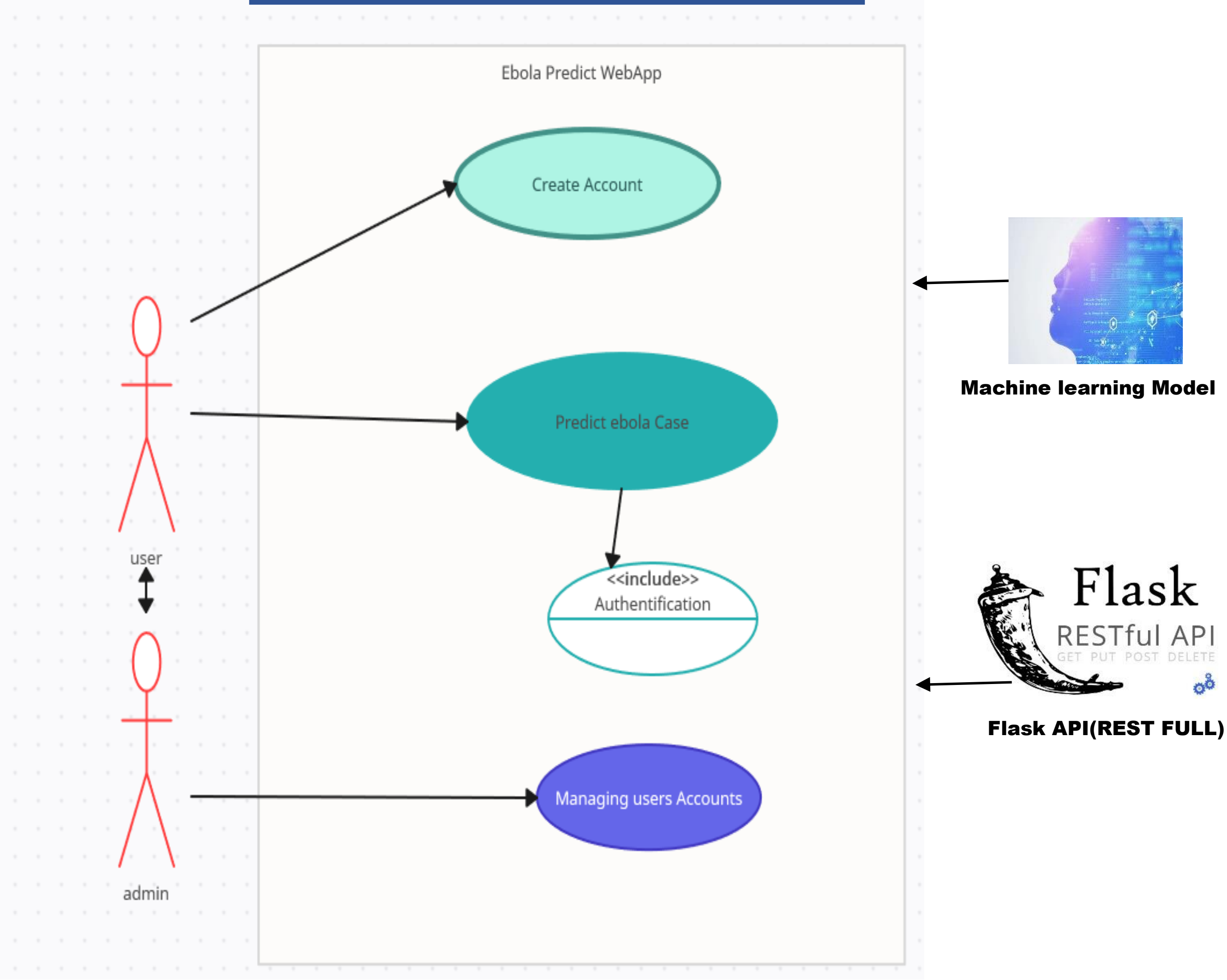
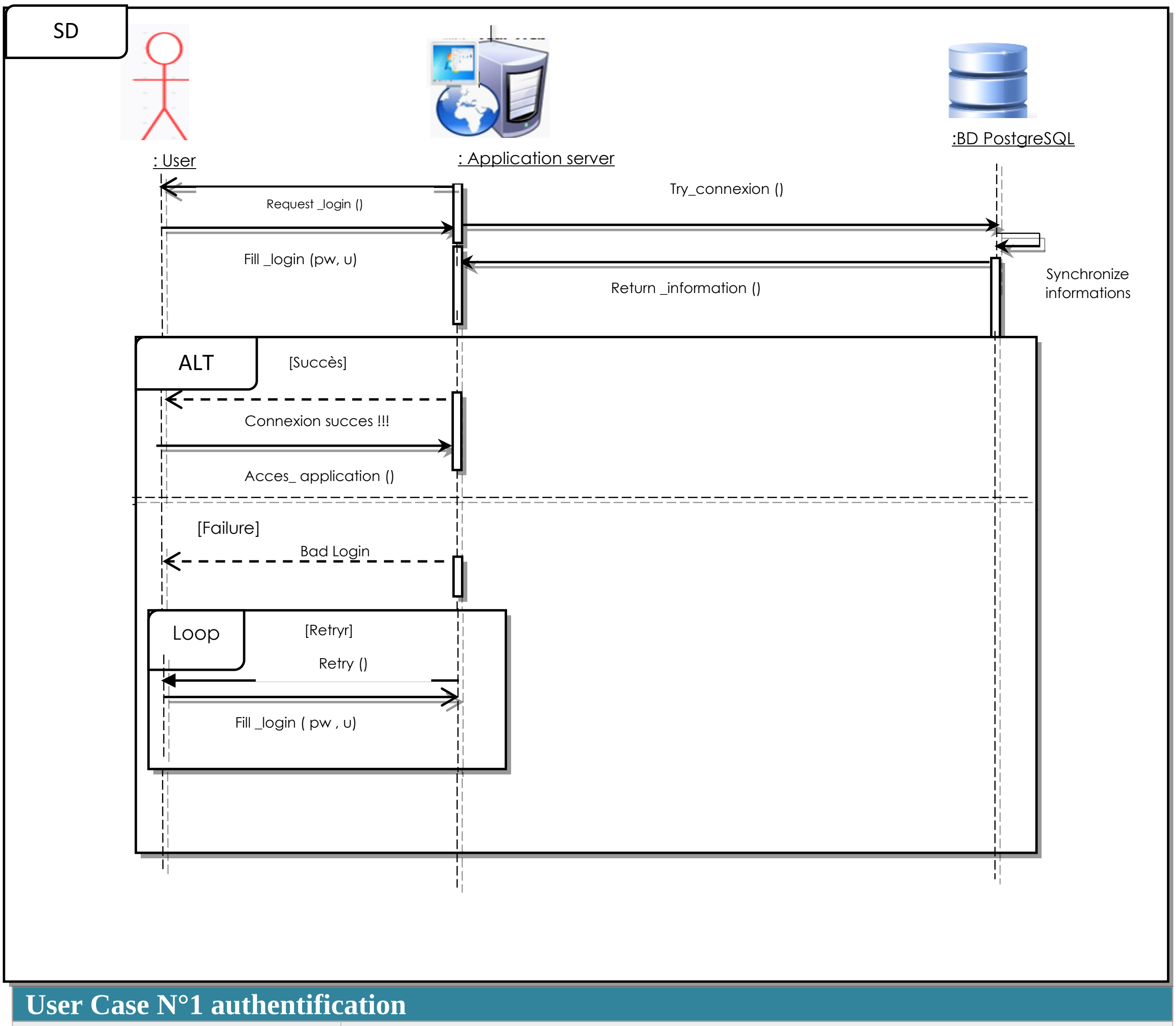


Figure 26:Use case Diagram

On this diagram, the ellipses represent system features (or cases of use), and each good guy is a system actor. The system is represented by the large rectangle. The actors are linked by the taches they can perform in the system via lines without flèches; those with flèches define the relationships between the cases of usage. The flèches with the phrase "include" indicate that an instance of the source case includes the behavior described by the destination case

b. Sequence Diagram

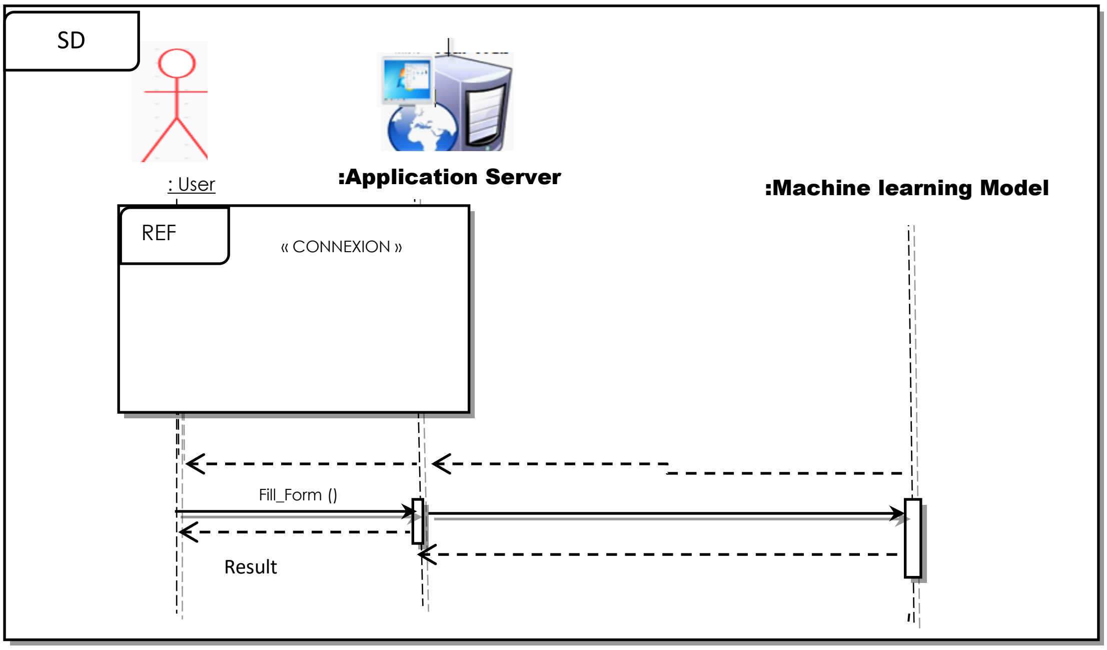
☞ Authentification



User Case N°1 authentification	
Actor	User, Administrator
But	This use case allows users to log into the app to access app features
Déclencheur	The app user
Information flow	Scenario : 1. The User enters the name and password; 2. The database management system verifies the information entered; 3. If the information is correct, the user accesses the home page, If the information is incorrect, the system returns an error message.
Extension	If the user does not have an account, he can create one

Figure 27:Sequece Diagram for Authentication

☞ Predict Ebola case



User Case N°2 Predict Ebola Case	
Actor	User, Administrator
But	this use case allows the user to predict a case of ebola by filling the form with symptoms
Déclencheur	The app user
Information flow	Scenario : 1. The User enters symptoms using form; 2. The system verifies the information entered; 3. if the information is correct, they are then sent to the model for the prediction

Figure 28:Predict Ebola, Sequence Diagram

j. Deployment diagram

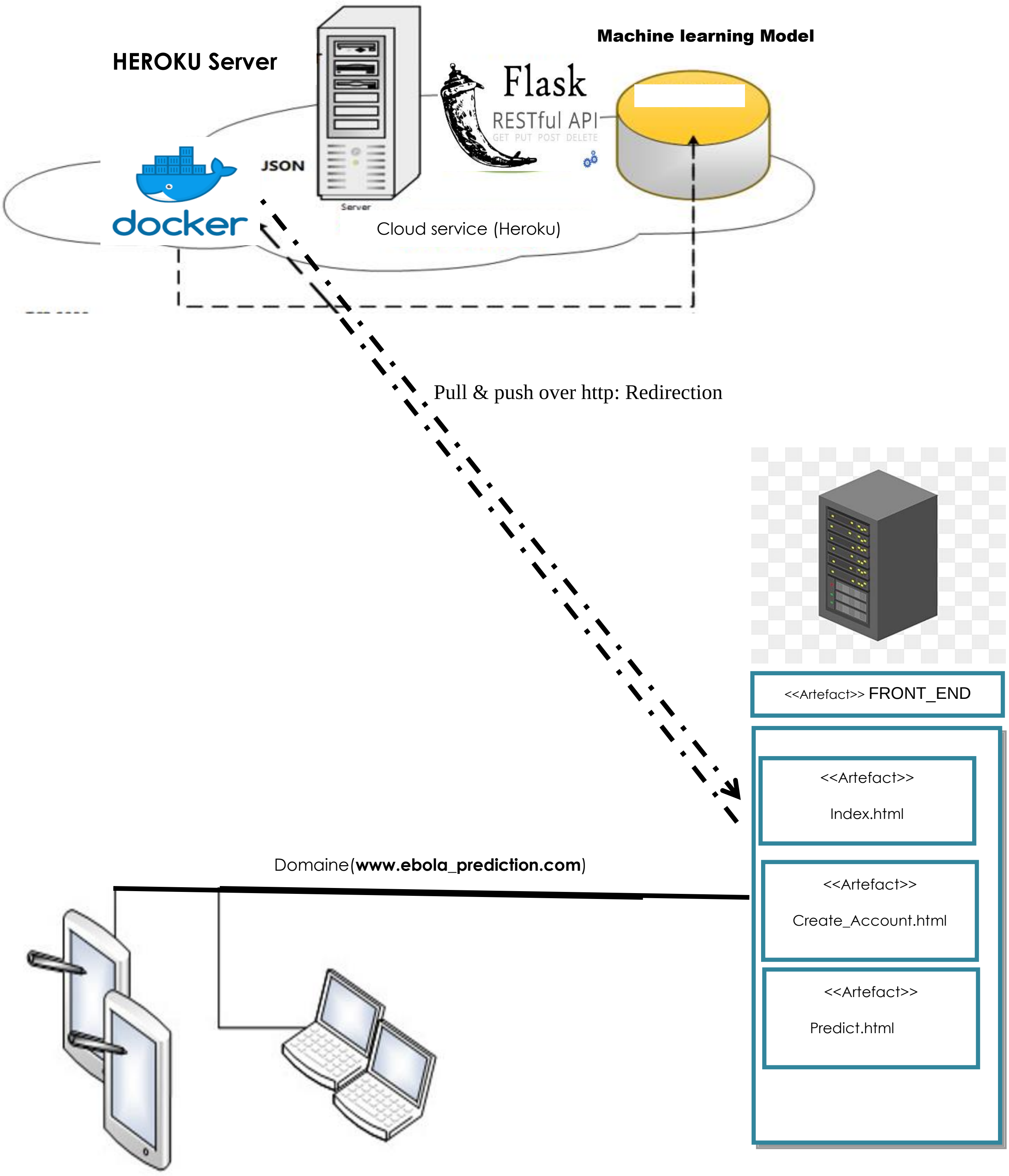


Figure 29:Deployment Diagram

Chapter IV : RESULTS FINDINGS

4.1 Model training and Evaluation

4.1.1 Evaluation of model

Model evaluation in machine learning is an integral part of building model because it helps to know the best model fit the data that will be used for future prediction. The evaluation of model performance is not done on the training for avoiding the problem of overfitting, but model evaluation uses test set (Etter, 2019)

Depending on the type of machine learning algorithms (classification or regression), there are numerous performance indicators used to evaluate machine learning models. Furthermore, it is preferable to use multiple evaluation metrics for a single model because one model may perform well with one evaluation metric and poorly with another, so using multiple evaluation metrics will help to determine whether the model performs correctly and optimally (Metrics, 2023).

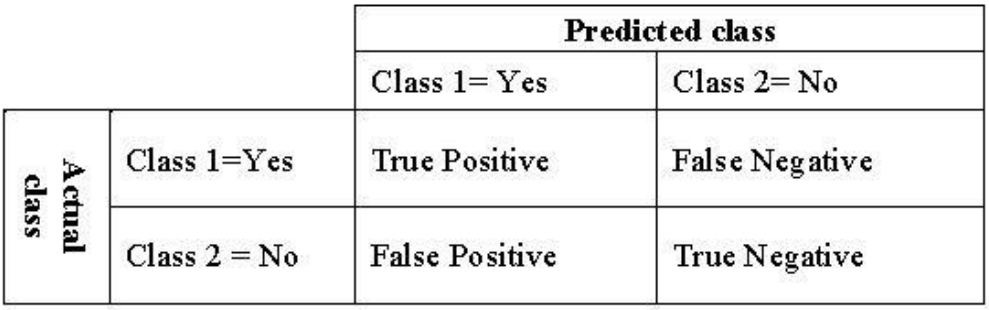
To measure the performance of the machine learning algorithm, the test set will be used in this study. Each machine learning algorithm is trained using the training set, and the evaluation of the machine learning algorithms is measured on the test set. The assessment measures employed in the study include Accuracy, Recall, Precision, F-score, and ROC. These evaluation metrics were used to measure the performance of six classifiers.

Table 2: Confusion matrix

True Negative tells us the number of cases that the classifier correctly predicted that the person is negative, for this study it indicates the total number of cases classified as no outbreak whereby no outbreak happened.

False Positives are also called Type 1 error. It tells us the number of cases that the classifier incorrectly predicted, it tells us the number cases predicted as outbreak whereby no outbreak happened (S. N. , 2023).

Table 3:Confusion matrix



		Predicted class	
		Class 1= Yes	Class 2= No
Actual class	Class 1=Yes	True Positive	False Negative
	Class 2 = No	False Positive	True Negative

From the confusion matrix, we can obtain evaluation metrics that will enable us to compare different machine learning models according to their performance. These metrics include :

- Accuracy
- Precision
- Recall
- F-score

a. Accuracy is known as the correctness of the model which is the sum of true predicted ebola outbreak over the total number of total ebola outbreak cases

$$Accuracy = \frac{TrueNegatives + TruePositive}{TruePositive + FalsePositive + TrueNegative + FalseNegative}$$

Equation 4: Accuracy formula. Source : (S. N. , 2023).

b. Precision is measure of accuracy of the correct ebola outbreak cases over the total ebola outbreak including correct classified and wrong classified.

$$Precision = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Positive(FP)}$$

Equation 5:Precision Formula. Source: (S. N. , 2023).

c. Recall is also called true positive rate is the measure of correct ebola outbreak cases over all cases should have been classified as ebola outbreak cases.

$$Recall = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Negative(FN)}$$

Equation 6: Recall Formula. Source: (S. N. , 2023).

d. F- score is the combination of Recall and precision of the model, it is considered as the harmonic mean of recall and precision of the model.

$$F_1\text{—score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

Equation 7:F-Score Formula. Source : (S. N. , 2023).

- Receiver Operating Characteristics (ROC) curve and Area under Curve (AUC)

ROC is defined as the plot of the test of sensitivity or true positive which is plotted on Y-axis versus the 1-specifity or the false positive on X-axis. It has been efficient to evaluate the performance or the quality of diagnostic test and mostly it is used in radiology test. ROC has a good performance through the decrease of standard error and as the number of test sample and Area Under Curve (AUC) increases as well as increase sensitivity when the analysis of variance.

4.1.2 Features Selection

One of the key approaches for increasing the chances of success in tackling machine learning challenges is feature engineering . Feature learning (also known as representation learning) is a technique used in feature engineering to create new features from a dataset. Let's look at how this strategy is applied. Before using the entire dataset as input for the machine learning algorithm, the Decision Tree machine learning algorithm was used to pick the key attributes for predicting ebola outbreaks.

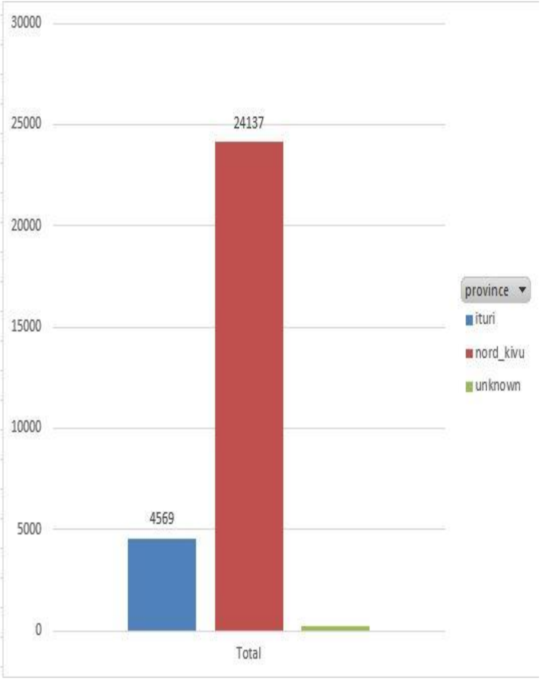
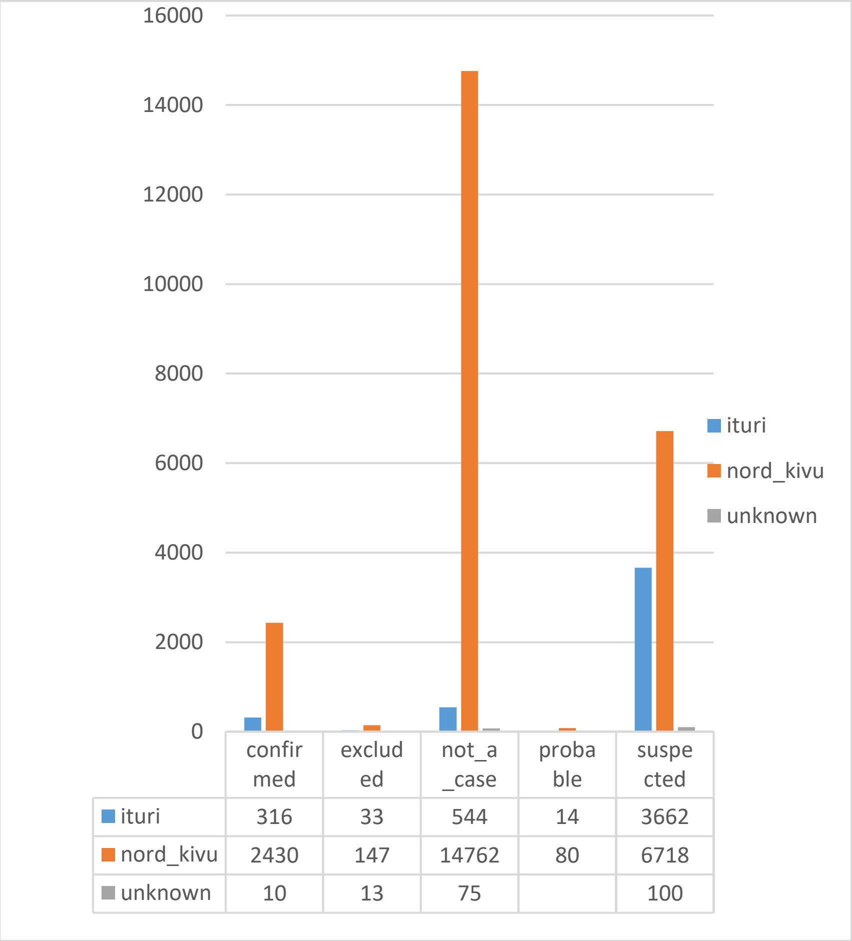


Figure 30:Ebola repartition case In NORTH-KIVU and Ituri Provinces



	confirmed	excluded	not_a_case	probable	suspected
🟦 ituri	316	33	544	14	3662
🟧 nord_kivu	2430	147	14762	80	6718
⬜ unknown	10	13	75		100

Figure 31:Ebola repartition case In NORTH-KIVU and Ituri Provinces using diagnostic

Table 4:Representation of the various attributes

Representation of the various attributes	
Age	*Age of the patient*
Province	*Province of the patient :Nord-kivu:1 ; ituri :2*
Gender	*Gender of the patient ; Male:1 ,female: 2*
chestpain	*chest pain ;1 :YES ; 2 : NO*
hematemesis	*SPIT BLOOD; 1:YES; 2:NO*
anorexia	*lack of appetite; 1:YES;NO:2*
unexplainedbleeding	*other bleeding;1:YES ;NO:2*
fever	*Fever;1 :YES ; 2 : NO*
vomiting	*Vomiting;1 :YES ; 2 : NO*
bleedgums	*gums bleeding;1 :YES ; 2 : NO*
diarrhea	*Diarrhea;;1 :YES ; 2 : NO*
bleedinject	*blood transfusion ;1 :YES ; 2 : NO*
fatigue	*Fatigue ; 1:YES; 2:NO*
bleednose	*nose bleed; 1:YES; 2:NO*
abdpain	*abdominal pain*
musclepain	*muscle pain ;1 :YES ; 2 : NO*
chestpain	*Chest pain;1 :YES ; 2 : NO*
bloodvomit	*vomit blood;1 :YES ; 2 : NO*
musclepain	*Muscle pain ; 1 :YES ; 2 : NO*
bloodcough	*blood cough; 1 :YES ; 2 : NO*
jointpain	*articular pain ; 1 :YES ; 2 : NO*
headache	*Headache ; 1 :YES ; 2 : NO*
bleedskin	*bleed from the skin; 1 :YES ; 2 : NO*
cough	*Cough ; 1 :YES ; 2 : NO*
bleedurine	*Bleed urine; 1 :YES ; 2 : NO*
diffbreathe	*breathe differently ; 1 :YES ; 2 : NO*
DIAG	*Diagnosis of the disease*

the experimental results are compiled and explained in detail in this section. The main database was filtered to get the test cases. This was done to overcome the problem of missing values for many attributes for a particular case. This resulted in the formulation of 28,904 test cases and 25 attributes. 1 and 2 are taken as the classifiers for diagnosis. The study evaluated the various symptoms as attributes in the database. The attributes used are described below.

4.1.3 Comparison of Accuracy of the existing Classification techniques:

The various classification techniques used in this research provided different accuracies when they were applied to the test data. The lowest accuracy was shown by Random Forest Classifier with 84.2% accuracy. Support Vector Classification showed 88% accuracy. The accuracy of each classification technique is listed below

Table 5:Comparison of Accuracy of the existing Classification techniques

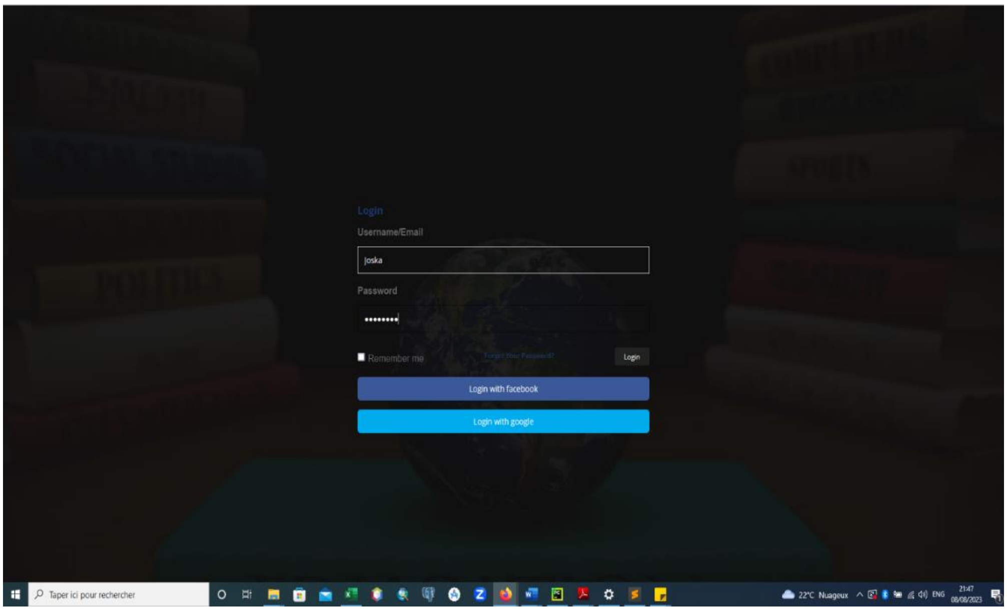
Classifier	Precision	Recall	F score	Accuracy
Support Vector Classification	0.88	0.99	0.93	0.88
Gradient Boosting Classifier	0.92	0.94	0.93	0.87
Decision Tree classifer	0.93	0.90	0.92	0.846
Random Forest Classifier	0.85	0.85	0.92	0.842

According to the table, among these all classifiers, Support Vector presents good performance metrics with more 88% of accuracy, precision, recall and F the score which means it can make 89% correct predictions, it is followed by Gradient Boosting with more 87% of accuracy and F score, recall and precision. The Decision Tree also presents a good performance metric where all their performance metrics are around 84,6% that make it among the good binary classifiers. The least performer model is Random Forest algorithms with 84.2% of accuracy, 0.92% of F score, 85.1% of recall and 85% on precision. In general, these machine learning algorithms perform well because most of them can make more 70% good classification prediction.

4.2 Deployment

In fulfillment of the second objective of this research, a web-based application was developed from which predictions of Ebola outbreak can be made. The application is a simple form that asks questions and gives an estimate of ebola case detection. The best models , is saved in pickle serialized format and then deployed within this flask-based application using Docker..

- Login page



Login

Username/Email: Joska

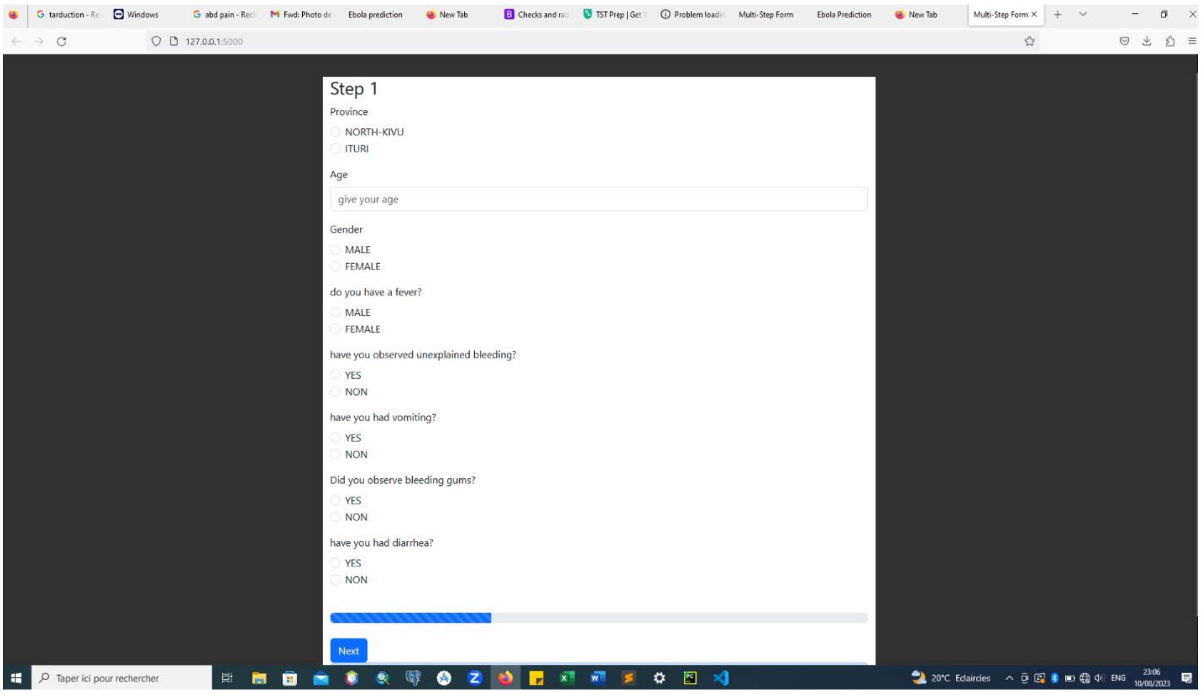
Password: ••••••••

□ Remember me

Figure 32: login form

on this form, the user enters his username and his password, and if they are correct, he will have access to the prediction form.

- Prediction Form



Step 1

Province
○ NORTH-KIVU
○ ITURI

Age
give your age

Gender
○ MALE
○ FEMALE

do you have a fever?
○ MALE
○ FEMALE

have you observed unexplained bleeding?
○ YES
○ NON

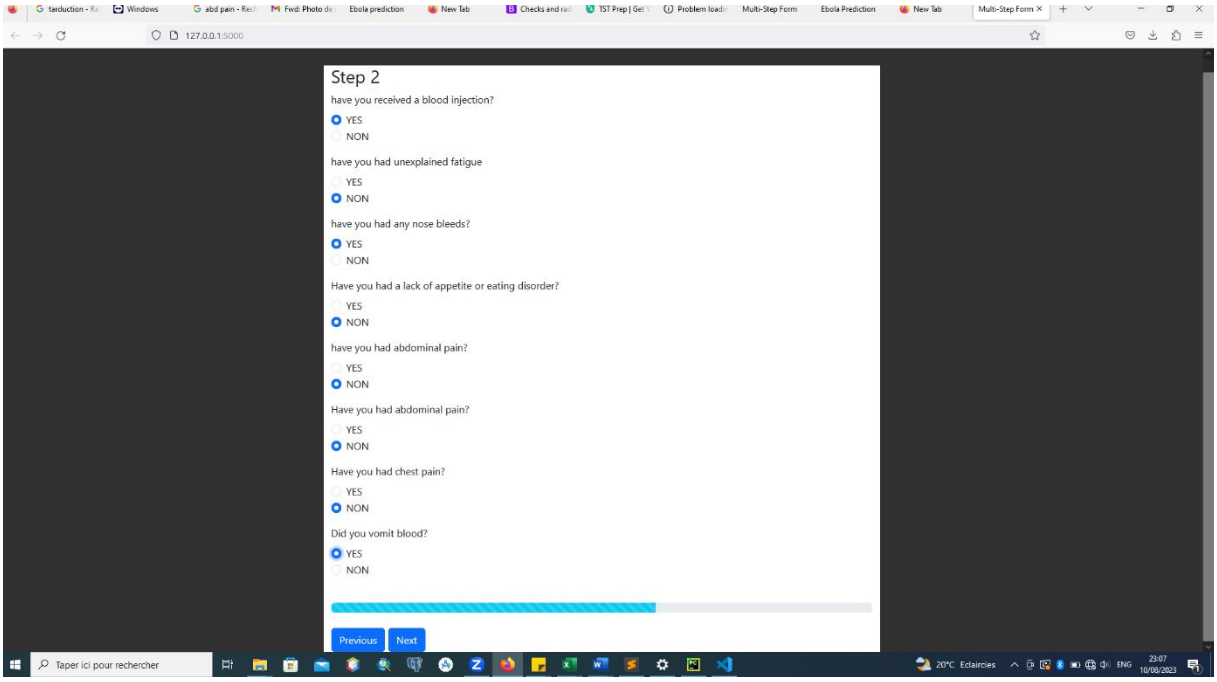
have you had vomiting?
○ YES
○ NON

Did you observe bleeding gums?
○ YES
○ NON

have you had diarrhea?
○ YES
○ NON

Figure 33:predict form 1

On this form, the user fills in a series of planned questions (consisting of ebola epidemic symptoms) and clicks on the next button to access the portion of questions that follow.



Step 2

have you received a blood injection?

● YES
○ NON

have you had unexplained fatigue

○ YES
● NON

have you had any nose bleeds?

● YES
○ NON

Have you had a lack of appetite or eating disorder?

○ YES
● NON

have you had abdominal pain?

○ YES
● NON

Have you had abdominal pain?

○ YES
● NON

Have you had chest pain?

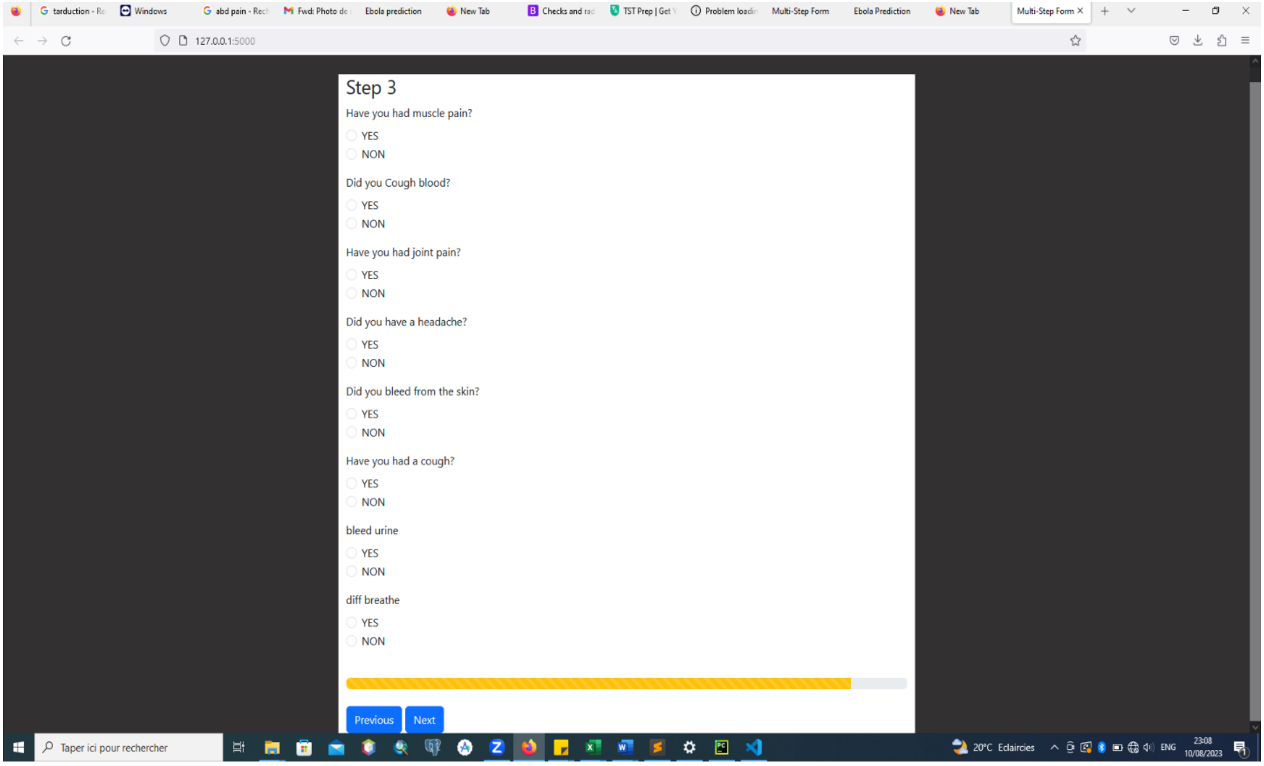
○ YES
● NON

Did you vomit blood?

● YES
○ NON

Figure 34:predict form 2

We do the same thing on this form as we did on the previous because the questions are many.



Step 3

Have you had muscle pain?

○ YES
○ NON

Did you Cough blood?

○ YES
○ NON

Have you had joint pain?

○ YES
○ NON

Did you have a headache?

○ YES
○ NON

Did you bleed from the skin?

○ YES
○ NON

Have you had a cough?

○ YES
○ NON

bleed urine

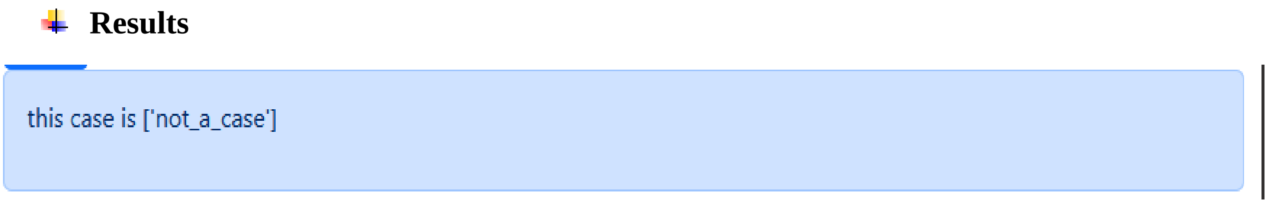
○ YES
○ NON

diff breathe

○ YES
○ NON

Figure 35: predict form 3

on this form, we do the same thing as on the second form to close and launch the prediction

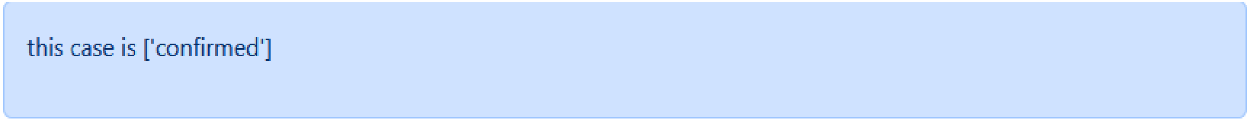


Results

this case is ['not_a_case']

Figure 36:Ebola response prediction(negative)

A response displayed when the prediction artificial intelligence is launched



this case is ['confirmed']

Figure 37:Ebola response prediction(positive)

Another confirmation response of an Ebola case

CHAPTER V : CONCLUSION AND RECOMMANDATIONS

5.1 CONCLUSION

For this study, we have built the prediction of Ebola outbreak by using symptoms shown by the patient during the early stages of the disease , the total of 28,904 records from Ebola outbreak cases from two provinces have been used. The study has also used machine learning algorithms especially classifiers, we have assessed quatre classifiers including Random Forest, Decision tree, Gradient Boosting, and Support vector machine.

The performance of these classifiers is evaluated using accuracy, Recall, Precision, F-score. Among the classifiers, Support vector Classification comes with high performance compared to other classifiers with more than 88% in all evaluation metrics; it has shown the accuracy of 90.75%, F-score of 93.7%, The precision of 88.1% and Recall of 99.08%. However, the other classifiers also have shown the high performance because they scored above 70% on the evaluation metrics. This high performance explained the linkage between Ebola and the symptoms shown by the patient during the early stages of the disease but also it has shown the efficiency of machine learning in prediction especially in classification.

This system should be used by hospitals, health care providers, health involved organizations, to be aware ahead of time whether there might happen the Ebola outbreak so that they can take precautions and make available the resources ahead of time so that the human lives be saved.

It is a contribution to the public health, and it might be used as one of Ebola control system so that burden caused by ebola should decrease as we use this model in the correct way.

This system should be used by any; hospitals, health care providers, health involved organizations thought our web application which will be hosted just by completing the patient's symptoms, the system predicts the outcome.

We were unable to build a USSD system to access the system with a telephone that does not have internet access for people and structures far from the city and not having smart phones because the internet and smartphone penetration rate in the DRC remains low; while specifying that we have built an API in Python for our web application.

Moreover, this study has only covered only 2 provinces because we were unable to collect data from other previously Ebola-affected provinces, if It was possible to access the data of the whole the country for areas affected by Ebola, would give the opportunity to train a big dataset which might increase the performance of algorithms so that the models would be more precise and accurate.

According to the results as displayed and discussed, the prediction of Ebbola outbreak has been successful based on machine learning algorithms of classification so by using the data, we have in the study and Random Forest algorithm, we are sure at more than 88% to make the correct prediction of Ebola outbreak. Therefore, I would like to call different health organizations to adopt and use it as one of the control and mitigation measures of Ebola outbreak, so that we can achieve the global target of Ebola’s eradication in DRC with technology.

Bibliography

visual-paradigm. (2023, 08 23).*what-is-uml.* Retrieved from www.visual-paradigm.com: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/

A., C., T., S., T., F., K., R., B., F., & P., S. (2018). Transforming clinical data into actionable prognosis models: machine-learning framework and field-deployable app to predict outcome of Ebola patients. *PLoS neglected tropical diseases,*, 3,10.

artoftesting. (2023, 8 28). *uml-in-software-engineering.* Retrieved from https://artoftesting.com/uml-in-software-engineering: https://artoftesting.com/uml-in-software-engineering

BD, D., MSY, L., Tiffany A, & al. (2018). Unreported cases in the 2014–2016 Ebola epidemic: Spatiotemporal variation, and implications for estimating transmission. *PLoS Negl Trop Dis*, 12.

Biswal, A. (2023, 05 26). *artificial-intelligence-tutorial*. Retrieved from Simplilearn : https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/types-of-artificial-intelligence

Brown, C. S., & Johnson, O. (2018). Introduction to Viral Haemorrhagic Fevers. In Ebola Virus Disease. *Springer, Cham.*, 1-7.

Brownlee, J. (2020). *Supervised and Unsupervised Machine Learning Algorithms.* Toronto: Guiding Tech Media.

Brownlee, J. (2023, 11 05). *Develop k-Nearest Neighbors in Python From Scratch*. Retrieved from Develop k-Nearest Neighbors in Python From Scratch: https://machinelearningmastery.com/tutorial-to-implement-k-nearest-nearestneighbors-/

Burns, E., Nicole Laskowski, & Linda Tucci,. (2023, 05 19). *AI-Artificial-Intelligence*. Retrieved from techtarget.com: https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence

Cheng, Y. Y., P. P. K. Chan, & Z. W. Qiu. (2012). Random forest based ensemble system for short term load forecasting,”. *Proc. - Int. Conf. Mach. Learn. Cybern., vol. 1, doi: 10.1109/ICMLC.2012.6358885.,* 52-56.

Chowell, G., L, S., Bansal S, & Viboud C. (2016). Mathematical models to characterize early epidemic growth: A review. Phys Life Rev . *science direct*, 66–97.

Colubri, A., Mary-Anne Hartley, Matthew Siakor, Vanessa Wolfman, August Felix, Tom Sesay, . . . Pardis C. Sabeti. (2019). Machine-learning Prognostic Models from the 2014–16 Ebola Outbreak: Dataharmonization. *elsevier*, 54–64.

disease, E. v. (2023, 06 26). *World Health Organization Regional Office for Africa. Health topics*. Retrieved from World Health Organization Regional Office for Africa: http://www.afro.who.int/health-topics/ebola-virus-disease.

Dudzik, S. (2012). Application of the naive Bayes classifier to defect characterization using active thermography. *Journal of Nondestructive Evaluation,*, 383-392.

Escott, E. (2023, 05 26). *codebots.com*. Retrieved from codebots: https://codebots.com/artificial-intelligence/the-3-types-of-ai-is-the-third-even-possible

Etter, P. (2019). Model evaluation. Underw Acoust Model. *doi:10.4324/9780203475652*, 261-278.

F, K., Gsteiger S, Low N, Hansen CH, & Althaus CL. (2016). Heterogeneity in District-Level Transmission of Ebola Virus Disease during the 2013–2015 Epidemic in West Africa. *PLoS Negl Trop Dis*, 10,7.

Fenn, B. (2023, 07 31). *Benefits of using UML in agile development.* Retrieved from Control Engineering: https://www.controleng.com/articles/benefits-of-using-uml-in-agile-development/

G., M., Varatharajan R., & Priyan M. K. (2018). Hybrid recommendation system for heart disease diagnosis based on multiple kernel learning with adaptive neuro-fuzzy inference system. *Multimedia tools and applications*, 4379-4399.

Goldman, P. (2013, 05 19). *How does generative AI work?* Retrieved from salesforce.com: https://www.salesforce.com/news/stories/what-is-generative-ai/?d=7010M0000025ltFQAQ&nc=&utm_source=google&utm_medium=paid_search&utm_campaign=amer_sfnewsaw&utm_content=what-is-generative-ai_7010M0000025ltFQAQ&utm_term=GenerativeAI&gclid=Cj0KCQjwmZejBhC_AR

Goss, E. P., & Ramchandani, H. (1995). Comparing classification accuracy of neural networks, binary logit regression. *Journal of Economics and Finance*, 1,3.

H, B. (2018, 05 23). *Excitement over use of Ebola vaccine in outbreak tempered by real-world challenges. Stat. Health.* Retrieved from https://www.statnews.com: https://www.statnews.com/2018/05/23/ebola-vaccine-drc-real-world-challenges/

J., B. M., Brooks T. J., & Pollock N. R. (2018). Diagnosis of Ebola virus disease: past, present, and future. *Clinical microbiology reviews, 29(4),*, 773-793.

javatpoint. (2023, 05 26). *javatpoint*. Retrieved from www.javatpoint.com: https://www.javatpoint.com/machine-learning

JD, K., Barrie MB, Mesman AW, & al. (2018). Anatomy of a hotspot: chain and seroepidemiology of Ebola virus transmission, Sukudu, Sierra Leone, 2015–16. J Infect . *google scholar*.

K., C., & Chhabra A. (2019 September). Evaluating the performance of tree based classifiers using Ebola virus dataset. *International Conference on Next Generation Computing Technologies (NGCT)*, 494-499.

Khatri, V. S. (2023, 08 23). *python-frameworks*. Retrieved from hackr.io: https://hackr.io/blog/python-frameworks

Kotsiantis, S. B., Zaharakis, I.,, & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 3-24.

kubernetes. (2023, 08 02). *kubernetes.io*. Retrieved from kubernetes.: https://kubernetes.io/

KURAMA, V. (2023, MAI 23). *Gradient Boosting In Classification: Not a Black Box Anymore*. Retrieved from Gradient Boosting In Classification: Not a Black Box Anymore: https://blog.paperspace.com/gradient-boosting-for-classification/

Lin, W., Z. Wu, L. Lin, A. Wen, & J. Li. (2017). An ensemble random forest algorithm for insurance big data analysis. *doi:10.1109/ACCESS.2017.2738069.IEEE Access vol. 5*, 16568–16575.

M., A., Lane T., Zorn K. M., Lingerfelt M. A., Clark A. M., & Freundlich J. S. (2019). Ebola Virus Bayesian Machine Learning Models Enable New in Vitro Leads. *ACS omega 4(1),*, 2353-2361.

M., M. (2020). Africa ' s Malaria Epidemic Predictor : Application of Machine Learning on Malaria Incidence and Climate Data. *google scholar*, 29-37.

Maimon, O., & Rokach, L. (2005). *Data mining and knowledge discovery handbook.* Eds.

ME, F. (2017). A Theory-Based Socioecological Model of Communication and Behavior for the Containment of the Ebola Epidemic in Liberia. *J Health Commun*, 22.

Metrics, E. (2023, 07 23). *DeepAI*. Retrieved from DeepAI: https://deepai.org/machine-learning-glossary-and-terms/evaluation-metrics

microfocus. (2023, 05 31). *microfocus*. Retrieved from https://www.microfocus: https://www.microfocus.com/en-us/what-is/machine-learning

Munappy, A. R., Jan Bosch, Helena Holmström Olsson, & Anders Arpteg. (september 2022). Data management for production quality deep learning models: Challenges and solutions. *Journal of Systems and Software*, 191.

Naik, N. (2017). Docker container-based big data processing system in multiple clouds for everyone. *IEEE*, 3.

Navlani, A. (2022, 11 04). *Understanding Logistic Regression in Python*. Retrieved from Understanding Logistic Regression in Python: https://www.datacamp.com/community/tutorials/understanding-logistic-regression-python

Nelson, D. (2023, MAI 23). *Gradient Boosting Classifiers in Python with Scikit-Learn,*. Retrieved from Gradient Boosting Classifiers in Python with Scikit-Learn,: https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-LEARN

O, A. A., & G. Jimoh R. (2018). Comparative Study on the Prediction of Symptomatic and Climatic based Malaria Parasite Counts Using Machine Learning Models. *Int J Mod Educ Comput Sci:doi:10.5815/ijmecs.2018.04.03*, 18-25.

Organization, W. H. (2018, 05 14). *Emergencies preparedness, response*. Retrieved from World Health Organization: http://www.who.int/csr/don/14-may-2018-ebola-drc/en/.

Organization, W. H. (2022, 11 12). *Situation on the current outbreak in North Kivu (2018-2019)*. Retrieved from Situation on the current outbreak in North Kivu : in North Kivu (2018-2019)" [online]. Available at: https://www.who.int/ebola/situation-reports/drc-2018/en/

Organization, W. H. (2022, 12 11). *World Health Organization*. Retrieved from ebola-virus-disease: https://www.who.int/news-room/fact-sheets/detail/ebola-virus-disease

P., G. E., & Ramchandani H. (1995). Comparing classification accuracy of neural networks, binary logit regression and discriminant analysis for insolvency prediction of life insurers. *Journal of Economics and Finance*, 1,3,19.

Pedamkar, P. (2023, 07 31). *data mining method.* Retrieved from educba.: https://www.educba.com/data-mining-methods/

Prevention, C. f. (2023, 08 23). *Centers for Disease Control and Prevention.* Retrieved from www.cdc.gov: https://www.cdc.gov/mmwr/volumes/68/wr/mm6850a3.htm

pythonbasics. (2023, 08 02). *what-is-flask-python.* Retrieved from pythonbasics: https://pythonbasics.org/what-is-flask-python/

Robinson, A. (2023, 05 15). *How to Calculate Euclidean Distance*. Retrieved from How to Calculate Euclidean Distance: https://sciencing.com/how-to-calculate-euclidean-distance-12751761.html.

S, F., Ciglenecki I, Tiffany A, & al. (2017). The impact of control strategies and behavioural changes on the elimination of Ebola from Lofa County, Liberia. *Philos Trans R Soc Lond B Biol Sci*, 372.

S., N. (2023, 07 23). *Towards Data Science.* Retrieved from Understanding Confusion Matrix: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

S., S., & Mangat, V. (september 2017). Relevance Vector Machine classification for big data on Ebola outbreak. *1st International Conference on Next Generation Computing Technologies (NGCT) IEEE*, 639-643.

Samuel, A. (2023, 05 31). *notebook.community*. Retrieved from https://notebook.community/: https://notebook.community/vravishankar/vravishankar.github.io/machine-learning/Introduction%20to%20Machine%20Learning

SEMAJERI, E., & UMEZURUIKE CHINECHEREM. (november 2019). System Analysis of an Informatics Model for Ebola Virus detection in Adults. *IRJET*, 2.

Shubham. (2022, 11 04). *Decision Tree*. Retrieved from Decision Tree: https://www.hackerearth.com/practice/machine-learning/machine-learning-algorithms/mlmldecision-

Stojiljković, M. (2022, 11 04). *Logistic Regression in Python*. Retrieved from Logistic Regression in Python: https://realpython.com/logistic-regression-python

tackoverflow. (2023, 08 02). Retrieved from tackoverflow: https://insights.stackoverflow.com/survey/2020#technology-most-loved-dreaded-and-wanted-languages-loved

Team, W. E. (2016). After Ebola in West Africa—Unpredictable Risks, Preventable Epidemics. N Engl J Med. *PubMed/NCBI,* 87–96.

turing. (2023, 08 02). *deploy-ml-models-with-flask-and-docker.* Retrieved from www.turing.com: https://www.turing.com/kb/deploy-ml-models-with-flask-and-docker

tutorialsteacher. (2023, 08 02). *Learn Python Programming.* Retrieved from .tutorialsteacher: https://www.tutorialsteacher.com/python

tutoriel, P. (2023, 08 23). *what-is-flask-python.* Retrieved from Python tutoriel: https://pythonbasics.org/what-is-flask-python/

US, C. W. (2019). *Ebola outbreak In DRC: Second-Largest Outbreak in History Rages in Congo.* https://www.concernusa.org/story/ebola-outbreak-in-drc/.

V., S. A. (2016). Intelligent Classification Systems. In Search Techniques in Intelligent Classification Systems. *Springer*, 1-13.

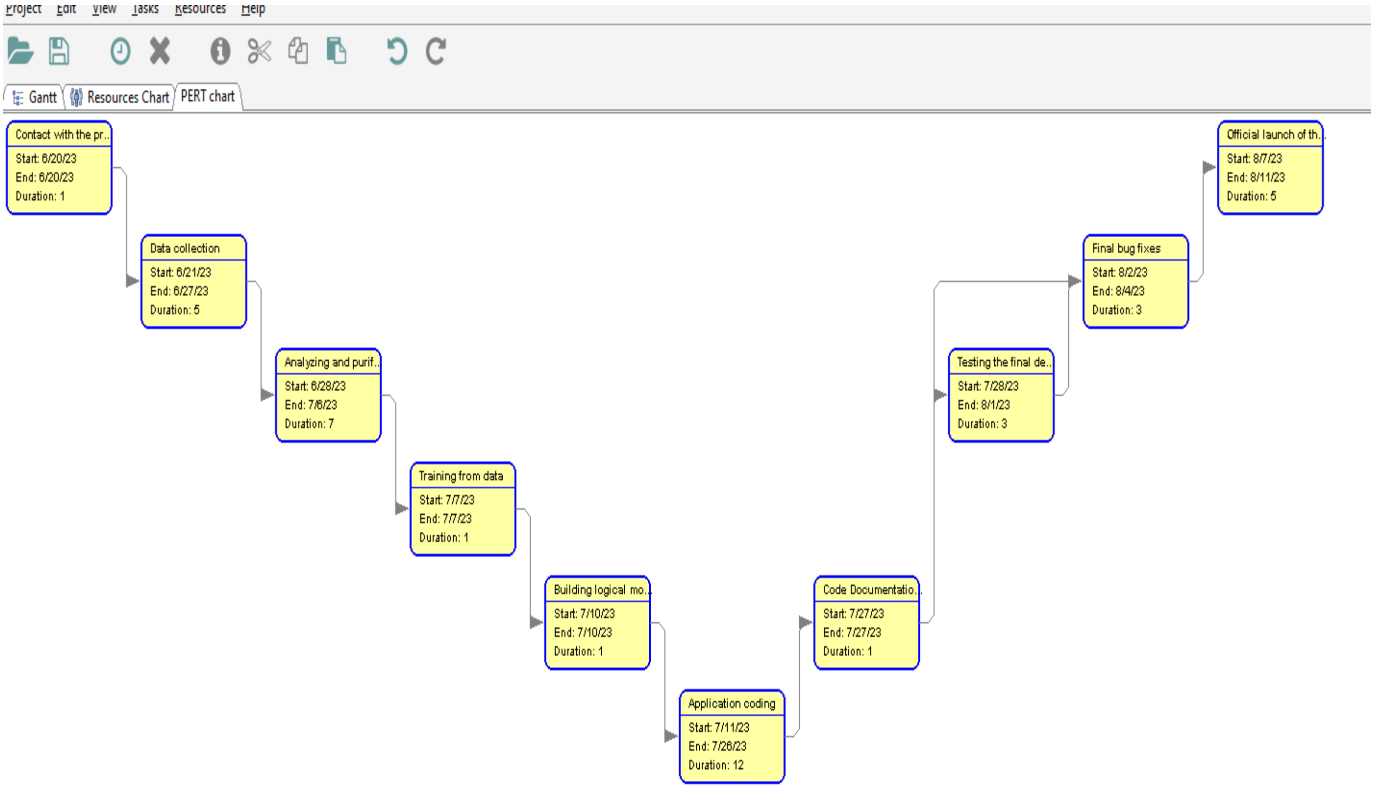
Wrike. (2023, 07 31). *what-is-pert-in-project-management/.* Retrieved from Wrike: https://www.wrike.com/project-management-guide/faq/what-is-pert-in-project-management/

Yan, T., Mu, J., E., Q., Wang, Y, Liu, L, Wu, D., & Li, J. (2015). Clinical characteristics of 154 patients suspected of having Ebola virus disease in the Ebola holding center of Jui Government Hospital in Sierra Leone during the 2014 Ebola outbreak. *European Journal of Clinical Microbiology & Infectious Diseases*, 2089-2095.

Zhang, P., B., C., Ma L., Li Z., Song Z., Duan W., & Qiu X. (2017). Relevance Vector Machine classification for big data on Ebola outbreak. *In 2015 1st International Conference on Next Generation Computing Technologies (NGCT)* (pp. 639-643). IEEE.

Appendix

PERT graph



Contact with the pr..
Start: 6/20/23
End: 6/20/23
Duration: 1

Data collection
Start: 6/21/23
End: 6/27/23
Duration: 5

Analyzing and purif..
Start: 6/28/23
End: 7/6/23
Duration: 7

Training from data
Start: 7/7/23
End: 7/7/23
Duration: 1

Building logical mo..
Start: 7/10/23
End: 7/10/23
Duration: 1

Application coding
Start: 7/11/23
End: 7/26/23
Duration: 12

Code Documentatio..
Start: 7/27/23
End: 7/27/23
Duration: 1

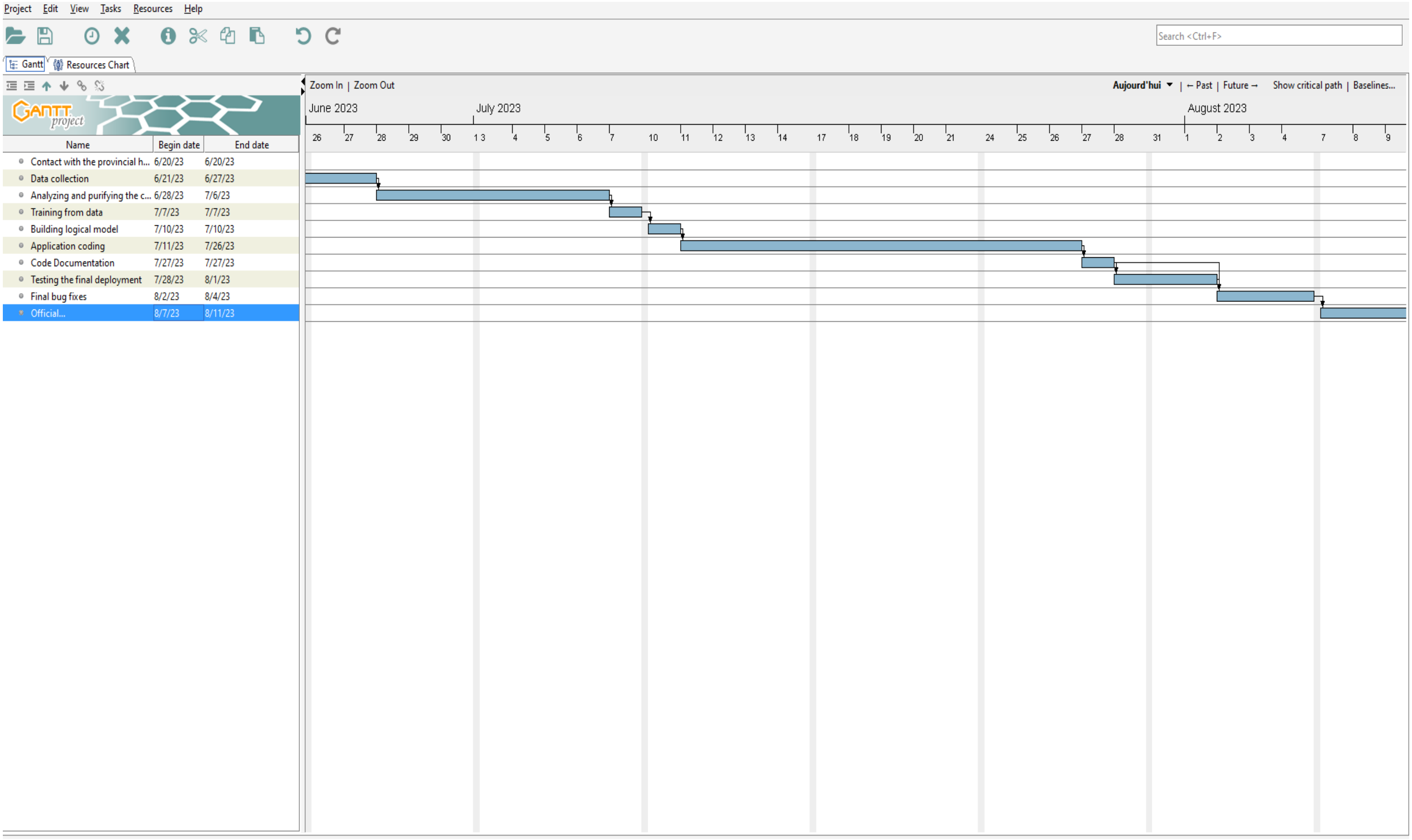
Testing the final de..
Start: 7/28/23
End: 8/1/23
Duration: 3

Final bug fixes
Start: 8/2/23
End: 8/4/23
Duration: 3

Official launch of th..
Start: 8/7/23
End: 8/11/23
Duration: 5

Figure 38:PERT graph of our project

GANTT chart



Name	Begin date	End date
Contact with the provincial h...	6/20/23	6/20/23
Data collection	6/21/23	6/27/23
Analyzing and purifying the c...	6/28/23	7/6/23
Training from data	7/7/23	7/7/23
Building logical model	7/10/23	7/10/23
Application coding	7/11/23	7/26/23
Code Documentation	7/27/23	7/27/23
Testing the final deployment	7/28/23	8/1/23
Final bug fixes	8/2/23	8/4/23
Official...	8/7/23	8/11/23

Figure 39: GANTT chart

- Python codes

```
import pandas as pd
col_names =
['province','age','gender','fever','unexplainedbleeding','vomiting','bleedgums','diarrhea','blee
dinject','fatigue','bleednose','anorexia','abdpain','hematemesis','chestpain','bloodvomit','mus
clepain','bloodcough','jointpain','headache','bleedskin','cough','bleedurine','diffbreathe',
'label']
# load dataset
pima = pd.read_csv("jos_fin_thesis.csv", header=None,
names=col_names,on_bad_lines='skip')

print(pima.head())   #5 premieres lignes
print(pima.shape)

#split dataset in features and target variable
feature_cols =
['province','age','gender','fever','unexplainedbleeding','vomiting','bleedgums','diarrhea','blee
dinject','fatigue','bleednose','anorexia','abdpain','hematemesis','chestpain','bloodvomit','mus
clepain','bloodcough','jointpain','headache','bleedskin','cough','bleedurine','diffbreathe']
X = pima[feature_cols] # Features
X=X.iloc[1:-1,:].values #Skipping first-line
y = pima.label # Target variable
y=y.iloc[1:-1].values

from sklearn.model_selection import train_test_split # Import train_test_split function
# Split dataset into training set and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=1) # 70%
training and 30% test


from sklearn.ensemble import HistGradientBoostingClassifier
# Import Decision Tree Classifier
# Create Decision Tree classifer object
clf = HistGradientBoostingClassifier()
# Train Decision Tree Classifer
clf = clf.fit(X_train,y_train)
#Predict the response for test dataset
y_pred = clf.predict(X_test)

# Model Accuracy, how often is the classifier correct?
from sklearn import metrics #Import scikit-learn metrics module for accuracy calculation
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))

from sklearn.metrics import confusion_matrix
confusion_matrix = confusion_matrix(y_test, y_pred)
print("confusio matrics:",confusion_matrix)
```

```
from sklearn.metrics import classification_report
print("report:",classification_report(y_test, y_pred))
```

Flask API

```
import pickle

import numpy as np
from flask import Flask,request,jsonify,render_template
from numpy import zeros, array

#create flask app
app=Flask(__name__)


#load a pikle file
model=pickle.load(open("ourModel.pkl","rb"))

@app.route("/")
def Home():
    return render_template("formulaire.html")

@app.route("/predict",methods =["POST"])
def predict():
    # int_features = [int(x) for x in request.form.values()]
    int_features = zeros(24, int)
    int_features[0] = request.form['province']
    int_features[1] = request.form['age']
    int_features[2] = request.form['gender']
    int_features[3] = request.form['fever']
    int_features[4] = request.form['unexplainedbleeding']
    int_features[5] = request.form['vomiting']
    int_features[6] = request.form['bleedgums']
    int_features[7] = request.form['diarrhea']
    int_features[8] = request.form['bleedinject']
    int_features[9] = request.form['fatigue']
    int_features[10] = request.form['bleednose']
    int_features[11] = request.form['anorexia']
    int_features[12] = request.form['abdpain']
    int_features[13] = request.form['hematemesis']
    int_features[14] = request.form['chestpain']
    int_features[15] = request.form['bloodvomit']
    int_features[16] = request.form['musclepain']
    int_features[17] = request.form['bloodcough']
    int_features[18] = request.form['jointpain']
    int_features[19] = request.form['headache']
    int_features[20] = request.form['bleedskin']
    int_features[21] = request.form['cough']
    int_features[22] = request.form['bleedurine']
    int_features[23] = request.form['diffbreathe']
    print(int_features)
    features = [np.array(int_features)]
    prediction = model.predict(features)
    print(prediction)
    return render_template("formulaire.html",prediction_text="this case is
{}".format(prediction))

if __name__ == "__main__":
```

```
    app.run(debug=True)
```

FONT-END Part (HTML,CSS,BOOTSTRAP)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <title>Multi-Step Form</title>
    <style>
        .form-step {
            display: none;
        }

        .form-step.active {
            display: block;
        }
    </style>
</head>
<body style="background-color:#323030; ">
    <div class="container mt-5">
        <div class="row justify-content-center">
            <div class="col-lg-8" style="background-color:#FFFFFF;">
                <form id="multi-step-form" class="needs-validation" novalidate method="post"
action="{{url_for('predict')}}">
                    <div class="form-step active" data-step="1">
                         <h3>Step 1</h3>
                        <div class="mb-3">
                            <label for="last-name" class="form-label">Province</label>

                            <div class="form-check">
  <input class="form-check-input" type="radio" name="province" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
   NORTH-KIVU
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="province" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
  ITURI
  </label>
</div>
                        </div>
                        <div class="mb-3">
                            <label for="middle-name" class="form-label">Age</label>
                            <input type="text" class="form-control" name="age" placeholder="give your age">
                        </div>
                        <div class="mb-3">
                            <label for="first-name" class="form-label">Gender</label>
                            <div class="form-check">
  <input class="form-check-input" type="radio" name="gender" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
   MALE
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="gender" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
  FEMALE
  </label>
</div>

                        </div>
                        <div class="mb-3">
                <label for="first-name" class="form-label">do you have a fever?</label>

<div class="form-check">
  <input class="form-check-input" type="radio" name="fever" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   MALE
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="fever" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
  FEMALE
  </label>
</div>

            </div>
            <div class="mb-3">
                <label for="first-name" class="form-label">have you observed unexplained
bleeding?</label>
                <div class="form-check">
  <input class="form-check-input" type="radio" name="unexplainedbleeding" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   YES
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="unexplainedbleeding" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
 NON
  </label>
</div>
            </div>
            <div class="mb-3">
                <label for="first-name" class="form-label">have you had vomiting?</label>




                            <div class="form-check">
  <input class="form-check-input" type="radio" name="vomiting" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   YES
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="vomiting" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
 NON
  </label>
</div>
            </div>
            <div class="mb-3">
                <label for="first-name" class="form-label">Did you observe bleeding gums?</label>

  <div class="form-check">
  <input class="form-check-input" type="radio" name="bleedgums" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   YES
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="bleedgums" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
 NON
  </label>
</div>

            </div>
            <div class="mb-3">
                <label for="first-name" class="form-label">have you had diarrhea?</label>
                  <div class="form-check">
  <input class="form-check-input" type="radio" name="diarrhea" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   YES
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="diarrhea" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
 NON
  </label>
</div>
                    </div>
                              <br>

                <div class="progress">
  <div class="progress-bar progress-bar-striped" role="progressbar" style="width: 30%" aria-valuenow="10" aria-
valuemin="0" aria-valuemax="100"></div>
</div>
 <br>
                    <button class="btn btn-primary next-step" data-step="2">Next</button>

                     <div class="alert alert-primary" role="alert">
 <p>this case is ['confirmed']</p>
</div>
                </div>

                <div class="form-step" data-step="2">
                    <h3>Step 2</h3>
                    <div class="mb-3">
                        <label for="last-name" class="form-label">have you received a blood injection?</label>

                        <div class="form-check">
  <input class="form-check-input" type="radio" name="bleedinject" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   YES
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="bleedinject" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
 NON
  </label>
</div>
                    </div>
                    <div class="mb-3">
                        <label for="middle-name" class="form-label">have you had unexplained fatigue</label>

                                    <div class="form-check">
  <input class="form-check-input" type="radio" name="fatigue" id="2" value="2">
  <label class="form-check-label" for="flexRadioDefault1">
   YES
  </label>
</div>
<div class="form-check">
  <input class="form-check-input" type="radio" name="fatigue" id="1" value="1">
  <label class="form-check-label" for="flexRadioDefault1">
 NON
  </label>
</div>
                    </div>
                    <div class="mb-3">
                        <label for="first-name" class="form-label">have you had any nose bleeds?</label>


         <div class="form-check">
  <input class="form-check-input" type="radio" name="bleednose" id="2" value="2">


    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
    <script>
        $(document).ready(function () {
```

```
      $('.next-step').click(function (e) {
        e.preventDefault();
        const currentStep = $(this).closest('.form-step');
        const nextStepNumber = parseInt($(this).attr('data-step'));
        const nextStep = currentStep.next(`[data-step="${nextStepNumber}"]`);
        currentStep.removeClass('active');
        nextStep.addClass('active');
      });

      $('.prev-step').click(function (e) {
        e.preventDefault();
        const currentStep = $(this).closest('.form-step');
        const prevStepNumber = parseInt($(this).attr('data-step'));
        const prevStep = currentStep.prev(`[data-step="${prevStepNumber}"]`);
        currentStep.removeClass('active');
        prevStep.addClass('active');
      });
    });
  </script>


<!-- Flexbox container for aligning the toasts -->


</body>
</html>
```