DEVELOPMENT OF PNEUMONIA PREDICTION SYSTEM USING CONVOLUTION NEURAL NETWORK (CNN)

A PROJECT REPORT

BY

OJO ADEBOWALE THEOPHILUS

(BIM/18/8531)

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF TECHNOLOGY (B. TECH) IN BIOMEDICAL TECHNOLOGY

AT THE

SCHOOL OF BASIC MEDICAL SCIENCE (SBMS)

DEPARTMENT OF BIOMEDICAL TECHNOLOGY,

THE FEDERAL UNIVERSITY OF TECHNOLOGY AKURE, ONDO STATE, NIGERIA

SUPERVISOR: PROF. OGIDAN

DECEMBER, 2023

# CERTIFICATION

This is to certify that this final year project titled " Development of pneumonia prediction system using convolution neural network (CNN)" was done by **OJO ADEBOWALE THEOPHILUS** with matriculation number **BIM/18/8531** in accordance with the regulations governing the award of Bachelor of Technology of the Department of Biomedical

Technology, School of Basic Medical Sciences, College of Health Sciences, Federal University of Technology, Akure.

DR O. K. OGIDAN …………………………………..

PROJECT SUPERVISOR SIGNATURE/DATE

DR V.A AKPAN …………………………………

HEAD OF DEPARTMENT SIGNATURE/DATE

# 

# DEDICATION

I dedicate the project to God almighty my creator, my strong pillar, and my source of imagination, wisdom, knowledge, and understanding. He has been the source of my strength throughout the program and on his wings only have I soared. I also dedicate this work to my lovely parents and my entire family who have encouraged me all the way and whose encouragement has made sure that I give it all that it takes to finish that which I have started. To my siblings who have been affected in every way possible by this quest. Thank you. My love for you all can never be quantified. Always keep your head up and God bless you.

# ACKNOWLEDEMENT

My deepest gratitude to God who has provided all that was needed to complete this project and the program for which it was undertaken for. I thank God for giving me the privilege of being a student of this great citadel of learning and for keeping me thus far. Throughout this entire study, he took care of everything that would have stopped me in my tracks and strengthened me even through my most difficult times. There are so many people I need to Acknowledge for their guidance and support in the completion of my degree. I wish to express my profound gratitude from the depth of my heart to my parents, MR and MRS OJO, my sister, and my brother, for their unending support through every up and down of my life, I pray that God will grant you long life in good health to reap the fruit of your tireless labor. I am indebted to you all always.

An exceptional appreciation for my amiable supervisor Dr. O. K Ogidan who committed his time to educate me about my research and also made necessary corrections to the research study. My appreciation also to goes to the other lecturers, Dr. S. I. Oyeleye, Dr. O. B. Ogunsuyi, and the non-academic staff of the Department of Biomedical Technology, the Federal University of Technology, Akure for the continuous support which all summed up to me becoming a Biomedical Technologist.

My appreciation also goes to my friends and colleagues in the Department for advice and word of encouragement this page can’t tell it all but all I could say is that you are all appreciated and God bless you all.

**TABLE OF CONTENTS**

[CERTIFICATION i](#_Toc156425379)

[DEDICATION ii](#_Toc156425380)

[ACKNOWLEDEMENT iii](#_Toc156425381)

[ABSTRACT iv](#_Toc156425382)

[LIST OF FIGURES v](#_Toc156425383)

[LIST OF PLATES vi](#_Toc156425384)

[LIST OF GRAPHS vii](#_Toc156425385)

[LIST OF TABLES viii](#_Toc156425386)

[CHAPTER ONE 1](#_Toc156425387)

[INTRODUCTION 1](#_Toc156425388)

[1.1 Overview of artificial intelligence 3](#_Toc156425389)

[1.3 AIM AND OBJECTIVES OF PROJECT 4](#_Toc156425390)

[1.3.1 AIM ……………………………………...……………………………….4](#_Toc156425390)

[1.3.2 OBJECTIVES ……………………………………...……………………….4](#_Toc156425390)

[1.4 Problem statement and scope of study 5](#_Toc156425391)

[1.4.1 Problem statement 5](#_Toc156425391)

[1.4.2 Scope of study 5](#_Toc156425391)

[1.5 Benefits of the design 6](#_Toc156425392)

[1.6 Limitation of the project 7](#_Toc156425392)

[CHAPTER TWO 9](#_Toc156425394)

2.0 [LITERATURE REVIEW 9](#_Toc156425395)

[2.1 Prediction Systems 9](#_Toc156425396)

[2.2 Types of prediction System 9](#_Toc156425397)

[2.3 Machine learning 10](#_Toc156425398)

[2.3.1 Types of machine learning 10](#_Toc156425399)

[2.3.2 How Machine Learning Works 12](#_Toc156425400)

[2.3.3 Application of Machine Learning 12](#_Toc156425400)

[2.4.1 Computer Vision 13](#_Toc156425400)

[2.4.2 Convolutional Neural Networks as a Technique for Deep Learning 14](#_Toc156425400)

[2.4.3 Components of the Convolutional Neural Networks 15](#_Toc156425400)

[2.4.4 Image processing 17](#_Toc156425400)

[2.4.5 Areas of Application………………………………………………………………….17](#_Toc156425400)

2.5 Pneumonia……………………………………………………………………19

[2.5.1 Pneumonia can be classified into three categories based on the American thoracic society 19](#_Toc156425413)

[2.5.2The most common organism causes pneumonia based on the above three categories 20](#_Toc156425414)

[2.5.3 Epidemiology of pneumonia in Nigeria 22](#_Toc156425414)

[2.6 Symptoms. 23](#_Toc156425415)

[2.6.1 Evaluation. 24](#_Toc156425415)

[2.6.2 Differential Diagnosis. 25](#_Toc156425415)

[2.6.3 Compilations. 25](#_Toc156425415)

[2.7 Summary of related research work 42](#_Toc156425416)

[CHAPTER THREE 48](#_Toc156425417)

[METHODOLOGY 48](#_Toc156425418)

[3.1 Collection of Pneumonia X-ray datasets 48](#_Toc156425419)

[3.1.1 Data Preprocessing 48](#_Toc156425420)

[3.2 Designing Machine Learning Models using MS WORD 51](#_Toc156425427)

3.2.1 Understanding MS WORD………………………………..51

3.2.2 Model Architecture …………………………………………………………………..51

[3.3 Implement the models using a TensorFlow 51](#_Toc156425428)

[3.3.1 CNN Algorithm 51](#_Toc156425428)

[3.4 Evaluate the models using binary Cross entropy loss and metrics such as Binary Accuracy, Precision, and Recall 59](#_Toc156425429)

[3.5 Deployment of the model on a web-based interface 61](#_Toc156425432)

[3.6 User Requirement 63](#_Toc156425434)

[3.6.1 Hardware Requirement …………………………………….……………….. 63](#_Toc156425434)

[3.6.1 Software Requirement …………………….…………………….…………... 64](#_Toc156425434)

[CHAPTER FOUR 65](#_Toc156425437)

[4.0 RESULTS 65](#_Toc156425438)

[4.2 The implementation System 65](#_Toc156425439)

[4.3 Display of system interface 66](#_Toc156425440)

[4.4 Output Page 67](#_Toc156425441)

[4.5 System Evaluation 69](#_Toc156425442)

CHAPTER FIVE ………………………………………………………………………………………………………………………………..83

5.1 DISCUSSION ………………………………………………………………………………………………………………..83

[CHAPTER SIX 86](#_Toc156425443)

5.0 CONCLUSION AND RECOMMENDATION……………………………………..…...86

[5.1 CONCLUSION 86](#_Toc156425444)

[5.2 RECOMMENDATIONS 86](#_Toc156425445)

[REFERENCES 88](#_Toc156425446)

[APPENDIX](#_Toc156425447)

# 

# LIST OF FIGURES

Figure 3.1: Preprocess image Page 50

Figure 3.2: Conv2D block diagram Page 53

Figure 3.3: convolution neural networks Page 55

Figure 3.4: CNN Architecture Page 57

Figure 3.5: Pneumonia Detection Flowchart Page 58

Figure 3.6: Flask Api block diagram Page 63

Figure 4.1: Home Page Page 66

Figure 4.2: interface predicted normal result Page 67

Figure 4.3: interface pneumonia predicted result Page 68

Figure 4.3: Model Evaluation Page 71

# LIST OF GRAPHS

Graphs 4.3: Model Evaluation Page 71

# 

# LIST OF TABLES

Table 2.1: Literature review Page 26

Table 3.1 hardware requirement Page 63

Table 3.2: software Requirement Page 64

Table 4.1 Evaluation of pneumonia detection system Page 70

**ABSTRACT**

This project focuses on developing an intelligent system capable of predicting pneumonia in individuals using Convolutional Neural Networks (CNN). Pneumonia, a severe lung infection, necessitates a swift and accurate detection method.

To build this system, I curated a diverse set of X-ray images featuring chests with and without pneumonia. Ensuring image quality and variability, I trained the system to recognize crucial patterns indicating pneumonia presence.

The project's significance lies not only in the creation of the system but also in the thorough evaluation of its performance. Employing metrics such as accuracy, precision, recall, and F1-score, we assessed its effectiveness not only on familiar images but also on entirely new ones.

After rigorous testing, the system demonstrated its ability to predict pneumonia in fresh X-ray images with the accuracy of 0.971. This endeavor exemplifies the application of intelligent technology to enhance healthcare, offering a faster and more reliable approach to pneumonia detection. It signifies a modest yet impactful stride toward leveraging artificial intelligence in addressing health challenges.

**CHAPTER ONE**

**1.1 INTRODUCTION**

In the 1950s, Alan Turing introduced the idea of machines that could learn and become smart on their own. This marked the beginning of machine learning. Over time, machine learning has been applied in different areas, such as improving security with face detection, making public transportation safer and more efficient, and recently, playing a significant role in healthcare and biotechnology (Siddiqui, *et al*. 2020). In recent times, there have been significant advancements in machine learning, bringing about progress and opportunities to ease the workload of physicians and enhance accuracy, prediction, and overall quality of care. Machine learning in healthcare acts as a valuable support tool for physicians and analysts, aiding them in fulfilling their roles, identifying healthcare trends, and creating models to predict diseases. In large medical organizations, machine learning is also applied to improve efficiency in managing electronic health records and detecting irregularities in blood samples, organs, and bones through medical imaging and monitoring. Additionally, it plays a role in robot-assisted surgeries (Tian, *et al*. 2020).

In healthcare, machine learning is applied to detect pneumonia, a respiratory infection affecting the lungs caused by bacteria, fungi, or viruses. Pneumonia leads to the formation of pus or fluid in the air sacs of the lungs, known as alveoli, resulting in breathing difficulties and pain for patients. Various factors contribute to pneumonia, including viruses, fungi, bacteria, and environmental factors like air pollution and sophisticated atmospheres. Additionally, undernutrition and the lack of safe water, sanitation, and basic health facilities are significant contributors.

Pneumonia has severe consequences, accounting for approximately 16% of the 5.6 million under-five deaths, causing the demise of around 880,000 children in 2016. According to WHO (2022), a child dies from pneumonia every 39 seconds, making it the leading cause of death among infectious diseases in children. Over 800,000 children under five lose their lives to pneumonia annually, equivalent to about 2,200 deaths every day, including over 153,000 newborns. Pneumonia is classified into infectious and non-infectious types, with infectious pneumonia further categorized into bacterial, viral, mycoplasmas, and chlamydial pneumonia. Non-infectious pneumonia is termed immune-associated pneumonia, and further distinctions are made based on medical pathogenesis, including CAP (community-acquired pneumonia), HAP (hospital-acquired pneumonia), and VAP (ventilator-associated pneumonia). Timely detection of pneumonia is crucial to preventing these tragic deaths.

The high number of pneumonia cases and its significant mortality rate have drawn the attention of researchers and scientists to develop effective detection methods. Various techniques have been employed for diagnosis, with radiology-based methods playing a crucial role. Chest X-rays, CT scans, and Magnetic Resonance Imaging (MRI) are utilized for pulmonary diseases. Among these, X-rays are the most popular and effective, available in almost every hospital. Larger medical centers offer options like CT scans and MRIs.

While X-rays are versatile and can detect various diseases, including pneumonia, distinguishing between different conditions becomes challenging. To enhance accuracy and enable earlier detection, advanced techniques are necessary. In recent years, Computer-Aided Designs (CAD) have gained prominence in the field of medical imaging. Machine learning and deep learning, especially Convolutional Neural Networks (CNN), have played a vital role in extracting useful features for image classification. Feature extraction is further refined through transfer learning methods, where CNN models leverage generic features from large-scale datasets. These advancements contribute to improved diagnostic capabilities in diseases such as breast cancer, lung nodules, mammograms, and others.

**1.2** **OVERVIEW OF ARTIFICIAL INTELLIGENCE**

Although the terms machine learning, deep learning, and artificial intelligence are typically used interchangeably, they represent different sets of algorithms and learning processes. Artificial Intelligence (AI) is the umbrella term that refers to any computerized intelligence that learns and imitates human intelligence. AI is most regarded for autonomous machines such as robots and self-driving cars, but it also permeates everyday applications, such as personalized advertisements and web searches.

In recent years, AI development and application have made incredible strides and have been applied to many areas due to their higher levels of decision-making, accuracy, problem-solving capability, and computational skills (Lee *et al.* 2017). In generally all development of AI algorithms, the data obtained is split into two groups, a training and test data set, to ensure reliable learning, representative populations, and unbiased predictions. As the name suggests, the training set is used for algorithm training that includes sets of characterizing data points features and corresponding predictions (in the case of supervised learning). The testing data set is new to the algorithm and is solely used to test the algorithm's abilities. This measure is taken to eliminate biases in the algorithm's testing by the training dataset (Alloghani *et al.* 2020). Once an algorithm passes through a training and testing phase with acceptable results, the algorithms are implemented in healthcare settings. The application of AI is broad and has many applied sub-regions; here, we provide an overview of machine learning and deep learning, two of the several sub-regions of AI.

Machine learning encompasses several different algorithmic models and statistical methods to solve problems without specialized programming. Several machine learning models are single-layered; therefore, large components of feature extraction and data processing are performed prior to inputting the data into the algorithm. Without the extra layers, these machine learning algorithms require intense data preprocessing in order for the algorithms to determine accurate predictions and to avoid over-fitting or under-fitting the training dataset. Deep learning is a more elaborate sub-form of machine learning that utilizes layered artificial neural networks and provides increased accuracy and specificity with decreased interpretability (Bhattacharjee, *et al.* 2017). The neuronal network method is characterized as the multilayer network that supports

the connection between the artificial neurons, or units, in each layer with that of the layer before and after it. These networks can learn, discern, and deduce from data on their own using these multilevel links for data processing, and the data are processed until the specialized results are achieved (Kamiński, *et al.* 2018).

**1.3** **AIM AND OBJECTIVES OF PROJECT**

The project aims to revolutionize pneumonia diagnosis through the development of a Pneumonia Prediction System using Convolutional Neural Network (CNN). This initiative is imperative to enhance early detection accuracy, ensuring timely intervention and improving patient outcomes. The specific objectives include creating a robust model capable of analyzing chest X-ray images efficiently, optimizing its performance, validating its reliability, and seamlessly integrating it into healthcare practices for widespread impact.

**1.3.1 AIM**

The project aims to develop a CNN-based pneumonia prediction system for healthcare

**1.3.2 OBJECTIVE**

1. To collect X-ray datasets
2. Design the machine learning models using MS word
3. Implement the models using TensorFlow
4. Evaluate the models using binary Cross entropy loss, Binary Accuracy, Precision, and Recall.
5. Deploy the model with the best performance using TensorFlow Lite
6. Evaluate the system based on users' perspectives

**1.4 PROBLEM STATEMENT AND SCOPE OF STUDY**

Highlighting the pressing issues within pneumonia diagnostics, this project aims to confront existing challenges and bridge gaps in current methods. By delving into the limitations of current approaches, the study focuses on the need for a more advanced and accurate predictive system. The scope encompasses a comprehensive evaluation of diagnostic inefficiencies, explores technological feasibility, and emphasizes the potential transformative impact on healthcare practices.

**1.4.1** **PROBLEM STATEMENT**

Pneumonia is a life-threatening respiratory infection that poses a significant global health concern, particularly affecting vulnerable populations such as children. The current methods for pneumonia detection, particularly through chest X-rays, face challenges in terms of accuracy and speed. Conventional diagnostic techniques may not be efficient enough for early and precise identification, leading to delays in treatment and potentially impacting patient outcomes. Therefore, there is a critical need for the development of an advanced and accurate pneumonia prediction system using Convolutional Neural Networks (CNN). This system aims to enhance the efficiency of pneumonia detection from chest X-ray images, providing healthcare professionals with a reliable tool for early diagnosis and intervention. The study seeks to address these challenges and contribute to the improvement of pneumonia detection methods, ultimately leading to more effective and timely patient care.

**1.4.2 SCOPE OF STUDY**

This study focuses on the application of machine learning techniques, specifically utilizing Convolutional Neural Networks (CNNs) implemented through TensorFlow, to detect and analyze pneumonia conditions from chest X-rays. The primary goal is to enhance the speed and accuracy of identifying pneumonia in medical imaging. Technologies employed in the implementation include TensorFlow, a widely-used machine learning framework, and Python as the programming language. The study involves the classification of chest X-ray images to distinguish between normal and pneumonia-affected cases. The nature of the classification results aims to provide timely and precise identification of pneumonia conditions.

Furthermore, the scope extends to the broader concept of integrating computers to assist healthcare professionals in decision-making processes. While machines are considered valuable aides, it is emphasized that they are not intended to replace medical practitioners. The project also holds potential significance for public health by aiding in early detection, particularly in regions with limited access to medical personnel.

**1.5** **BENEFITS OF THE DESIGN**

These models can assist healthcare professionals in accurately detecting and classifying various chest conditions. This can lead to earlier and more precise diagnoses, facilitating prompt and targeted treatment it can also process a large number of images quickly and accurately, allowing for faster turnaround times and cost savings in healthcare settings. By providing insights and predictions based on the analysis of chest X-ray data, these models can aid clinicians in making informed decisions regarding patient management, treatment plans, and follow-up protocols.

Timely identification of chest conditions through machine learning can lead to early intervention and prevention of disease progression. Detecting abnormalities and potential diseases at an early stage can improve patient outcomes, reduce morbidity and mortality rates, and minimize the need for invasive procedures or costly treatments.

By accurately identifying and prioritizing high-risk cases, machine learning models can help optimize the allocation of healthcare resources. Efficiently directing resources, such as specialized diagnostic procedures or consultations, to patients with the greatest need can improve resource utilization and enhance overall healthcare system efficiency. By also leveraging automated analysis, these models can bridge the gap in healthcare access and provide reliable diagnostic support to regions where access to radiologists or specialized clinicians is limited and it can also can assist in tailoring treatment plans and interventions to optimize outcomes for each patient.

**1.6 LIMITATION OF THE PROJECT**

While exploring the Application of Machine Learning for the Analysis and Diagnosis of Chest Conditions using Chest X-ray Data has numerous potential benefits, it also has some limitations. These limitations include:

1. Availability and Quality of Data: The success of machine learning models heavily relies on the availability and quality of the dataset. Obtaining a diverse and representative chest X-ray dataset with accurately labeled images can be challenging. Limited access to high-quality data or imbalanced class distributions may affect the model's performance and generalizability.
2. Interpretability of Model Decisions: Machine learning models, particularly deep learning algorithms, are often considered black boxes, making it difficult to interpret and explain their decisions. Understanding why a particular prediction was made by the model might be challenging, which can be a concern in critical healthcare applications where interpretability is crucial.
3. Ethical Considerations: As with any project involving medical data, ethical considerations must be addressed. Ensuring patient privacy, obtaining proper consent, and adhering to data protection regulations are essential. Additionally, potential biases in the dataset or model predictions must be carefully considered to avoid disproportionately affecting certain populations or perpetuating existing biases.
4. Generalization to Unseen Data: Machine learning models trained on specific datasets may struggle to generalize to new and unseen data. Models developed using chest X-ray data from a particular population or healthcare facility may not perform as well when applied to different populations or settings. Robustness and generalization should be carefully evaluated to ensure the model's reliability in various scenarios.
5. Expertise and Resources: Developing and implementing machine learning models for medical applications requires expertise in both machine learning and medical domain knowledge. Acquiring the necessary expertise, access to computational resources, and specialized infrastructure can pose challenges and increase project complexity.
6. Clinical Validation and Integration: Before applying machine learning models to real clinical settings, thorough validation and integration into existing healthcare systems are necessary. Validating the performance and safety of the models in collaboration with medical professionals is essential to ensure reliable results and successful translation of the technology into clinical practice.

**CHAPTER TWO**

**2.0 LITERATURE REVIEW**

**2.1 Prediction Systems**

Prediction systems are designed to make accurate forecasts or estimates about future events or trends based on historical data and statistical models. These systems combine prediction models with additional data and analytical tools to provide a comprehensive picture of the situation and help users make informed decisions. Human disease predication is a crucial part of human life. Early disease prediction of a human is an important step in the treatment of disease. Since the very beginning, a doctor has handled it almost exclusively. Thus, the healthcare industry thrives on innovation to make logistics efficient (Zhou *et al*. 2016). Innovation is the heart of the medical industry. It is what drives new treatments, cures and therapies (Littell *et al*. 1994). Innovation is also what keeps the medical industry current and relevant. The scope of development in the medical industry is vast (Milella *et al*. 2021). There are many areas where innovation is needed to make progress. Some of these include developing new treatments for diseases, finding ways to improve patient

care, and making medical procedures more efficient. In the current digital age, innovation in the medical industry can be achieved through the digitalization of medical processes (Kelly *et al*. 2017).

* 1. **Types of prediction system**

There are different types of prediction systems, like the following:

1. Time Series Models: These models are used to predict future values of a variable based on its past values.
2. Machine Learning Models: These models use algorithms to learn from historical data and make predictions.

**2.3 Machine Learning**

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. Machine learning encompasses several different algorithmic models and statistical methods to solve problems without specialized programming (Samuel 2000).

**2.3.1 Types of machine learning**

Machine learning model fall into four categories

1. Supervised machine learning

Supervised learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms to classify data or predict outcomes accurately. As input data is fed into the model, the model adjusts its weights until it has been fitted appropriately. This occurs as part of the cross-validation process to ensure that the model avoids overfitting or underfitting. Supervised learning helps organizations solve a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox. Some methods used in supervised learning include neural networks, naïve bayes, linear regression, logistic regression, random forest, and support vector machine (SVM).

1. Unsupervised machine learning

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. This method’s ability to discover similarities and differences in information make it ideal for exploratory data analysis, cross-selling strategies, customer segmentation, and image and pattern recognition. It’s also used to reduce the number of features in a model through the process of dimensionality reduction. Principal component analysis (PCA) and singular value decomposition (SVD) are two common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, and probabilistic clustering methods.

1. Semi-supervised learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of not having enough labeled data for a supervised learning algorithm. It also helps if it is too costly to label enough data.

1. Reinforcement machine learning

Reinforcement machine learning is a machine learning model that is similar to supervised learning, but the algorithm is not trained using sample data. This model learns as it goes by using trial and error. A sequence of successful outcomes will be reinforced to develop the best recommendation or policy for a given problem.

**2.3.2** **How Machine Learning Works**

1. A Decision Process: In general, machine learning algorithms are used to make a prediction or classification. Based on some input data, which can be labeled or unlabeled, your algorithm will produce an estimate about a pattern in the data.
2. An Error Function: An error function evaluates the prediction of the model. If there are known examples, an error function can make a comparison to assess the accuracy of the model.
3. A Model Optimization Process: If the model can fit better to the data points in the training set, then weights are adjusted to reduce the discrepancy between the known example and the model estimate. The algorithm will repeat this “evaluate and optimize” process, updating weights autonomously until a threshold of accuracy has been met.
   * 1. **Application of Machine Learning**
4. Speech recognition: It is also known as automatic speech recognition (ASR), computer speech recognition, or speech-to-text, and it is a capability which uses natural language processing (NLP) to translate human speech into a written format. Many mobile devices incorporate speech recognition into their systems to conduct voice search e.g. Siri or improve accessibility for texting.
5. Customer service: Online chatbots are replacing human agents along the customer journey, changing the way we think about customer engagement across websites and social media platforms. Chatbots answer frequently asked questions (FAQs) about topics such as shipping, or provide personalized advice, cross-selling products or suggesting sizes for users. Examples include virtual agents on e-commerce sites; messaging bots, using Slack and Facebook Messenger; and tasks usually done by virtual assistants and voice assistants.
6. Computer vision: This AI technology enables computers to derive meaningful information from digital images, videos, and other visual inputs, and then take the appropriate action. Powered by convolutional neural networks, computer vision has applications in photo tagging on social media, radiology imaging in healthcare, and self-driving cars in the automotive industry.
7. Recommendation engines: Using past consumption behavior data, AI algorithms can help to discover data trends that can be used to develop more effective cross-selling strategies. This approach is used by online retailers to make relevant product recommendations to customers during the checkout process.
8. Automated stock trading: Designed to optimize stock portfolios, AI-driven high-frequency trading platforms make thousands or even millions of trades per day without human intervention.
9. Fraud detection: Banks and other financial institutions can use machine learning to spot suspicious transactions. Supervised learning can train a model using information about known fraudulent transactions. Anomaly detection can identify transactions that look atypical and deserve further investigation.

**2.4.1 Computer Vision**

Computer Vision is the field of interpreting image content. It is concerned with the classification of the entire image, such as in a system classifying photos uploaded to the internet (Facebook, Instagram). Or Computer Vision is concerned with the recognition of objects in an image, such as detecting faces or car license plates (Facebook, Google Streetview). Or it is concerned with the detection of aspects of an image, such as cancer detection in biomedical images.

Classic Computer Vision (CV) algorithms unfortunately have their limitations and usually get the correct results with small deformations of objects, and under similar lighting conditions, which were associated with the building of a model set of characteristics describing an object. Widespread use of Convolutional Neural Networks (CNN) as a tool to aid the machines to make decisions based on a set of images of the surrounding world was possible through two processes. The first was the appearance of CPUs with high computing abilities, especially high-performance graphics cards processors which do very well in the network learning process. The second was the development of Machine Learning (ML) techniques, in particular Deep Learning (DL) algorithms.

Both ML and DL are concepts in the field of research of Artificial Intelligence (AI) or Computational Intelligence (CI).

Computer Vision has received new impetus by the use of so-called Deep Learning algorithms, with which one can classify decently large image collections. Google and Facebook are competing to provide the best interface to those learning algorithms: Google offers TensorFlow, Facebook provides PyTorch, both of which are Python-based libraries. Which is then proceed with a method that had been popular just before the arrival of Deep Learning algorithms, namely feature extraction and matching.

**2.4.2** **Convolutional Neural Networks as a Technique for Deep Learning**

DL is implemented by large-scale neural networks, which the best example are the deepest convolutional neural networks. According to (Goodfellow, *et al* 2016), Deep Learning is a section of Machine Learning, centered around algorithms modeling high-level abstraction in data sets, using multiple layers of nonlinear transformations. The most well-known and used group of deep learning algorithms are Convolutional Neural Networks because of their wide application possibilities in recognizing different patterns, particularly in detecting objects in digital images.

The idea of CNN has been around for a long time, but there were many problems that inhibited its development. For example, with the enlarging number of network’s layers, the number of model parameters increases, which, in conjunction with the simultaneous rise in the size of the input training base, significantly expands the demand for computing power. The vanishing gradient problem, associated with the use of the reverse propagation algorithm, also required a solution.

In recent years there has been a rapid development of CNN, and the impact of many difficulties has been significantly reduced as a result of research into new concepts for the broader neural network, including (He, *et al*. 2015). At the same time, there has been an increase in the ability to create very large datasets (Russakovsky, *et al*. 2015) based on a big data revolution.

**2.4.3** **Components of the Convolutional Neural Networks**

CNNs are inspired by the human visual system and are particularly well-suited for tasks that involve understanding spatial hierarchies and patterns in data.

In the construction of these networks, there are four main types of layers that perform the basic tasks of such networks:

1. convolution
2. pooling
3. normalization and
4. connection.

As a result of the Convolution Layer, the input image is processed by a variety of convolutional filters to extract the characteristics contained in those parts. Pooling Layer is being use for reduce the size of the information being analyzed, thereby decreasing the sensitivity of the network to the distortion of the analyzed scene. The basic methods used in this layer are max pooling, when the largest value is selected in the parsed window and averaging, when its value is averaged. The ReLED layer (Rectified Linear Units Layer) by data normalization increases the network’s ability to solve nonlinear problems. CNN consist of multiple layers on successive levels, but the last link in such a network is the submission of results to the final layer – Fully Connected Layer. This layer results in the final rating, allowing the various training categories assignment. The distinguishing feature of CNN over classical neural networks is that the number of layers is much higher. The depth of neural network architecture is defined as the length of the longest path between the input and output neurons. There is no precise threshold of the layers number, allowing one to call the network deep, but it has been assumed that it refers to the network with more than two hidden layers.

Several fields are related to Computer Vision, two of which are closely related, namely

1. Image Processing and
2. Machine Vision

Those two fields overlap with Computer Vision to such an extent, that their names are sometimes used synonymously. Here is an attempt to discriminate between them, although there exist no agreed definitions and distinctions:

**2.4.4** **Image processing**

Image Processing is concerned with the transformation or other manipulation of the image with the goal to emphasize certain image aspects, e.g. contrast enhancement, or extraction of low-level features such as edges, blobs, etc.; in comparison, Computer Vision is rather concerned with higher-level feature extraction and their interpretation for recognition purposes.There exist two other fields that overlap with Computer Vision

1. Pattern Recognition
2. Computer Graphics
3. Pattern Recognition (Machine Learning) is the art of classification (or categorization). To build a good computer vision system, it requires substantial knowledge of classification methodology. Sometimes it is even the more significant part of the computer-vision system, as in case of image classification, for which so-called Deep Neural Networks have produced the best classification accuracy.
4. Computer Graphics is sometimes considered as part of Computer Vision. The objective in Computer Graphics is to represent objects and scenes as compactly and efficiently as possible. however, there is no (semantic) recognition of any kind involved.
   * 1. **Areas of Application**
5. Medical imaging: registering pre-operative and intra-operative imagery; performing long-term studies of people’s brain morphology as they age; tumor detection, measurement of size and shape of internal organs; chromosome analysis; blood cell count.
6. Automotive safety: traffic sign recognition, detecting unexpected obstacles such as pedestrians on the street, under conditions where active vision techniques such as radar or lidar do not work well.
7. Surveillance: monitoring for intruders, analyzing highway traffic, monitoring pools for drowning victims.
8. Gesture recognition: identifying hand postures of sign level speech, identifying gestures for human computer interaction or teleconferencing.
9. Fingerprint recognition and biometrics**:** automatic access authentication as well as forensic applications.
10. Retrieval**:** as in image search on Google for instance.
11. Visual authentication: automatically logging family members onto your home computer as they sit down in front of the webcam.
12. Robotics: recognition and interpretation of objects in a scene, motion control and execution through visual feedback.
13. Cartography**:** map making from photographs, synthesis of weather maps.
14. Radar imaging: target detection and identification, guidance of helicopters and aircraft in landing, guidance of remotely piloted vehicles (RPV), missiles and satellites from visual cues.
15. Remote sensing**:** multispectral image analysis, weather prediction, classification and monitoring of urban, agricultural, and marine environments from satellite images.
16. Machine inspection: defect and fault inspection of parts: rapid parts inspection for quality assurance using stereo vision with specialized illumination to measure tolerances on aircraft wings or auto body parts; or looking for defects in steel castings using X-ray vision; parts identification on assembly lines.

**2.5 Pneumonia**

Pneumonia is an umbrella term for a group of syndromes caused by a variety of organisms that result in infection of the lung parenchyma. There have been many attempts to classify pneumonia based on the etiology, clinical setting in which the patent acquired the infection, and the pattern of involvement of lung parenchyma, among other classifications. Pneumonia is a severe form of acute lower respiratory infection that is responsible for high morbidity and mortality rates among children under five; it poses a major threat to public health worldwide (Liu, et al 2016). Globally, pneumonia is the leading cause of child mortality, responsible for approximately 6.0% of the 5.9 million deaths in the under-five age group, killing around 900,000 children in 2015. It accounts for the loss of over 2,500 children’s lives every day, or over 100 every hour (UNICEF 2016).

**2.5.1** **Pneumonia can be classified into three (3) categories based on the American thoracic society.**

1. Community acquired Pneumonia (CAP)

Any pneumonia acquired outside of a hospital in a community setting.

1. Hospital-Acquired Pneumonia (HAP)

Any pneumonia acquired 48 hours after being admitted in an inpatient setting such as a hospital and not incubating at the time of admission is considered as HAP. This classification helps clear the confusion surrounding the terms healthcare-associated and hospital-acquired pneumonia. Now all pneumonia acquired in the setting of assisted-living facilities, rehabilitation facilities, and other healthcare facilities have been included under community-acquired pneumonia, and a hospital setting is necessary for classifying pneumonia as HAP. (Kalil, *et al.* 2016)

1. Ventilator Associated Pneumonia (VAP)

Any pneumonia acquired 48 hours after endotracheal intubation is considered as VAP. (CCCTG 2006) These categories have helped establish the common organisms responsible for each type of pneumonia and helped to formulate treatment guidelines for the efficient management in both in-patient and out-patient setting. Depending on the pattern of involvement, pneumonia has historically also been studied as:

1. Focal non-segmental or lobar pneumonia: involvement of a single lobe of the lung.
2. Multifocal bronchopneumonia or lobular pneumonia
3. Focal or diffuse interstitial pneumonia. (Gharib. *et al.* 2001)

While identifying an etiologic agent for pneumonia is essential for effective treatment as well as epidemiological record keeping, this is seldom seen in clinical practice. (Bartlett, *et al.* 2011)

**2.5.2** **The most common organism causes pneumonia based on the above three (3) categories**

1. Community-Acquired Pneumonia
2. Bacterial causes

They have been classically studied under the subheadings "typical" and "atypical" organisms in terms of ease of culture positivity. Common typical organisms include Pneumococcus, Hemophilus influenzae, Moraxella catarrhalis, Group A Streptococcus, and other aerobic and anaerobic gram-negative organisms. Atypical organisms commonly seen in clinical practice include Legionella, Mycoplasma, Chlamydia, among others. (Sattar, *et al.* 2023) In the United States, the most common bacterial causes of CAP include Streptococcus pneumoniae, Staphylococcus aureus, Mycoplasma pneumoniae, and gram-negative enteric bacilli. (Jain, *et al*. 2015)

1. Viral causes

It is often observed that viral species colonize nasopharynx of patients with CAP. Whether they are the primary cause or contribute to the pathogenesis by secondary bacterial causes is still being investigated. However, some of the most frequent viral agents implicated in CAP in the United States include influenza virus followed by respiratory syncytial virus, parainfluenza virus, and adenoviruses. (Jain. *et al.* 2015)

1. Fungal causes

Fungal infections are usually implicated in patients with certain predisposing immunocompromised states like HIV and organ transplant recipients, among others. However, often overlooked, some fungal species can cause pneumonia in immunocompetent individuals which results in a delay in diagnosis and leads to unfavorable outcomes. The 3 commonest ones in North America include Histoplasma, Blastomyces, and Coccidioides. (Hage. *et al.* 2012)

1. Hospital-Acquired Pneumonia and Ventilator-Associated Pneumonia

There is considerable overlap in the etiologic agents in non-ventilated hospitalized patients and ventilated patients with pneumonia, and it is, therefore, appropriate to consider them together. These include:

1. Gram-negative bacilli like Escherichia coli, Pseudomonas Aerugenosa, Acinetobacter, and Enterobacter among others
2. Gram-positive cocci like Staphylococcus aureus; both Methicillin-sensitive and resistant, although the latter is more prevalent (Weiner. et al. 2016)
3. Other viruses and fungi that are more prevalent in immunocompromised and severely ill patients

**2.5.3** **Epidemiology of pneumonia in Nigeria**

The epidemiology of pneumonia in Nigeria reflects a significant public health challenge, particularly affecting children under the age of five and communities with limited access to healthcare. Pneumonia remains a leading cause of child mortality globally. (GBD 2016) Nigeria has the largest absolute number of annual pediatric pneumonia deaths globally with pneumonia accounting for 20% of under-5 deaths nationally. (Adewemimo *et al.* 2017) Northern Nigeria is reported as a clear hotspot for pneumonia mortality. (Reiner, *et al*. 2019) Pneumonia has historically been a leading cause of morbidity and mortality in the country. Several factors contribute to the prevalence of pneumonia, including malnutrition among children, limited access to healthcare facilities, and delays in seeking medical attention. Malnourished children are more susceptible to respiratory infections, exacerbating the impact of pneumonia.

Malnutrition is a key risk factor for poor pneumonia outcomes and premature death. (Sonego *et al*. 2015) Nigeria has one of the highest prevalence of childhood malnutrition in the Africa region, (Akombi *et al*. 2016) with an estimated 37% of children stunted and 9% malnourished and higher burden in the North. (NPC 2019) Demographic and social factors (e.g., crowding, poverty, low maternal education) are consistently identified as risk factors for childhood malnutrition, (Obasohan *et al.* 2020)

In Nigeria, pneumonia can be caused by various infectious agents, including bacteria, viruses, and fungi. Efforts to combat pneumonia have included vaccination campaigns targeting common causative agents such as Streptococcus pneumoniae and Haemophilus influenzae type b (Hib). Environmental factors, such as air pollution and exposure to indoor smoke from the use of solid fuels for cooking, also play a role in the incidence of respiratory infections. Public health initiatives have sought to address these factors, raising awareness, promoting vaccination, and improving overall access to healthcare services.

While progress has been made in vaccination efforts and public health awareness, ongoing challenges persist. Regular updates from reputable sources such as the World Health Organization (WHO) and the Nigeria Centre for Disease Control (NCDC) are crucial for understanding the current epidemiological landscape of pneumonia in Nigeria.

**2.6** **Symptoms**

Historically, the chief complaints in case of pneumonia include systemic signs like fever with chills, malaise, loss of appetite, and myalgias. These findings are more common in viral pneumonia as compared to bacterial pneumonia. A small fraction of patients may have an altered mental status, abdominal pain, chest pain, and other systemic findings. Pulmonary findings include cough with or without sputum production. Bacterial pneumonia is associated with purulent or rarely blood-tinged sputum. Viral pneumonia is associated with watery or occasionally mucopurulent sputum production. There may be an associated pleuritic chest pain with the concomitant involvement of the pleura. Dyspnea and a diffuse heaviness of the chest are also seen occasionally.

**2.6.1** **Evaluation**

Evaluation of CAP and HAP involves:

1. Clinical Evaluation

Involves performing a thorough history and physical examination

1. Radiological Evaluation

According to the Infectious Diseases Society of America (IDSA) and American Thoracic Society (ATS) guidelines, a demonstratable infiltrate by chest x-ray is necessary and is considered the best method (with supportive clinical findings) for the diagnosis of pneumonia. (Mandell, et al. 2007) Findings may vary from lobar to interstitial infiltrate, to occasionally cavitary lesions with air-fluid levels suggestive of a more severe disease process.

1. Laboratory Evaluation

These include a series of tests like blood culture, sputum culture and microscopy, routine blood counts, and lymphocyte count. Special tests such as urinary antigen testing, bronchial aspirate, or induced sputum may be used for certain pathogens. Two tests, procalcitonin and C-reactive protein help differentiate viral from bacterial causes when clinical and radiological findings may not be obvious. It is also noteworthy that empiric antibiotic treatment may be initiated in all typical cases of pneumonia, and the entire battery of tests is seldom needed. (Mandell, *et al.* 2007)

Evaluation of VAP, on the other hand, is a bit different from that of CAP. It requires radiological and microbiological evidence prior to initiation of antimicrobial therapy. VAP should be suspected in ventilated patients who have new onset dyspnea, fall in oxygen saturation on the same ventilator settings, fevers with chills or new onset lung infiltrates. All suspected patients require a chest x-ray (or a CT scan if x-ray findings are inconclusive). This must be followed by invasive sampling techniques like mini broncho-alveolar lavage (BAL) or bronchoscopic BAL or even protected specimen brush (PSB) to identify causal organisms. Once the diagnosis is confirmed, the appropriate antimicrobial therapy can be initiated. (Torres, *et al*. 2017)

**2.6.2** **Differential Diagnosis**

Differential diagnosis of pneumonia includes asthma, chronic obstructive pulmonary disease (COPD), pulmonary edema, malignancies, non-infective consolidative processes of the lung, pleuritis, pulmonary embolism, aspiration of a foreign body, bronchiectasis, bronchiolitis, and others just to name a few. In case a differentiation becomes difficult, parameters like C-reactive protein, erythrocyte sedimentation rate, procalcitonin levels, leucocyte count, and temperature may be used to establish a diagnosis. (Castro-Guardiola, *et al*. 2000)

**2.6.3 Complications**

Complications of untreated or under-treated pneumonia include respiratory failure, sepsis, metastatic infections, empyema, lung abscess, and multi-organ dysfunction. (Sattar, et al. 2023)

**Table 2.1:** **Literature review**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **WORK TITLE** | **PROBLEM SOLVED** | **METHODOLOGY** | **UNIQUENESS** | **LIMITATION** | **SOFTWARE** |
| Heart Disease Prediction Using Logistic Regression | logistic regression model in the early detection of cardiovascular diseases | The machine learning model was implemented in a Python 3.7 environment, utilizing ML libraries such as scikit-learn, pandas, and matplotlib, with Jupiter Notebook used for code execution. | The logistic regression (LR) model achieved an accuracy of 87.10%. | Its limitation being capable of classifying a single case of disease. | Python  Jupyter notebook |
| Logistical regression and random forest for classification of heart disease. | To classify heart diseases. | Logistic regression | The model achieved an accuracy of 85.04% | Its limited to a single case of disease. | Python |
| International evaluation of an AI system for breast cancer screening. | Deep learning algorithm to detect tumors based on Mammograms in earlier stages of growth | Deep learning model | The deep learning-based approach was able to outperform experienced radiologists by an AUC score of 11.5. | The model can only identify tumor. It cannot be generalized to different cases | Tensorflow  Python |
| Deep learning for healthcare decision making with EMRs  (Electronics Health Records) | Diagnose chronic Prediction Using Medical Notes.  To predict onset of diseases, such as heart failure, kidney failure and stroke. | LSTM (Reinforcement learning) used for remember things, make decisions based on what they remember, and even figure out when to change their decisions.  CNN (Supervised learning) Is the deep learning model used for extracting features. | Use both structured data obtain from EHR and unstructured data contained in progress and diagnoses notes.  The model used can be generalize to different diseases  Model performance (AUC) by Target Disease  Heart failure: 0.900  It’s as three main types of input data    Raw test of medical notes, lab  vital sign data recovered from medical notes  structured demographic data | Due to unavailability of well annotated data the model is limited to heart, kidney and stroke conditions.  LSTM model is becoming traditional which still limit the training of the model and to obtain higher accuracy for some certain conditions.  Due to lack of an annotated structured data the model accuracy (AUC) is less for certain disease.  1. Stroke: 0.833  2. Kidney failure: 0.733 | * TensorFlow framework * Keras * Python * Convolutional NN * LSTM |
| CheXNet: Radiology-level Pneumonia Detection on Chest X-rays with Deep Learning | To detect Pneumonia from chest X-rays | cheXNet is a 121-layer Dense Convolutional Network (DenseNet) to detect pneumonia at the level that exceeding Human Level Performance (Radiologist) | The model was built with Heatmap to identify the area of the image most indicative of pneumonia.  The CheX-Net trained on 112,120 frontal-view chest Xray images individually labelled with up to 14 different thoracic diseases. Including pneumonia  The model performance on the detection of pneumonia is beyond human level performance | Diagnose only frontal view X-ray  Limited to structured data alone | Convolutional Neural  tensorflow |
| Dermatologist-level classification of skin cancer with deep neural networks. | To classify appearance of skin lesion | Deep CNN model is the model built to classify the appearance of this disease.  The function of the CNN is to extract feature from the input data  At the output layer of the model was logistic regression model which classify the disease  Trained on 129, 450 dermatologist-labelled clinical images (datasets) | The model utilizes a GoogleNet inception v3 CNN architecture that was pretrained on approximately 1.28 million images (1000 object categories) which was trained on 129,450 dermatologist dataset using transfer learning. This makes the model to generalize well to prediction of diseases.  The deep learning CNN outperforms the average of the dermatologist at skin cancer classification using photographic and dermoscopic images in which the CNN was tested against at least 21 dermatologists at keratinocytes carcinoma and melanoma recognition | Image Data  Cannot classify many visual conditions but can be adapted to other specialties, including ophthalmology, otolaryngology, radiology and pathology if train with sufficient image data. | Deep CNN  GoogleNet |
| A hybrid machine learning framework to predict mortality in paralytic ileus patients using electronic health records (EHRs) | To help intensivist, surgeons and other medical professionals to optimize clinical management for PI patients in terms of optimal treatment strategy and resources planning. | The model has two phases:  Phase 1: statistical testing which uses Cox regression (univariate and multivariate) to detect the patient hazard ratio  Phase 2: machine learning application and in this application five (5) different machine learning algorithms to test the validity of the Framework.  The effectiveness for each model was evaluated by comparing accuracy, sensitivity, specificity, Receiver operating Characteristic (ROC) curves, and area under the curve (AUC).  A total of 1021 patients’ data were extracted from the original dataset of the MIMIC III database that were diagnosed with PI to trained and test the effectiveness of the model. | The uniqueness of this model is that it uses Cos Regression with Kaplan-Meier analysis to obtain the risk factors which is the hazard ratio that occur in phase one and this phase improve the prediction capabilities  The phase II reduces the time complexity of the predictive models by reducing the number of risk factors that needed to be processed by the machine learning. | Using two phases to build this model required a lot of processes and which waste a lot resources.  The training data (Risk factor) is insufficient likewise the testing data which makes the model to be prone to overfitting  The accuracy obtained during the training of this model was 81.30% which is not good for deployment.  Using Deep Learning approaches with TensorFlow framework will increase the accuracy and often make deployment easier. | Deep CNN  Tensorflow  Keras |
| Predicting post-stroke pneumonia using deep neural network approaches. | Deep neural network to timely and accurate predict post-post stroke pneumonia in clinical practice. | 13,930 acute ischemic stroke (AIS) patients EHR data from 2007-2017 were used to build and evaluate the model.  The data was divided into two datasets for training and testing.  In the design of the model three classic method were compared i.e.,  logistic regression (LR) which estimates the parameters of a logistic model in form of binomial regression.,  support vector machine (SVM)  use kernel functions such as radial basis kernel functions and linear kernel functions to project high-dimensional samples into lower dimensions to improve the prediction or classification ability of the model,  XGBoost (extreme gradient boosting) and two deep neural network methods, i.e., multi-layer perceptron (MLP) and GRU (gated recurrent unit). | One of the uniqueness of this model is that it based on large set of risk factors and can detect earlier risk patients and they are able to provide a clear time course of pneumonia after AIS that is the model was built for pneumonia patients within specific time window and after onset of stroke.  The model can leverage useful information in the full range of the patient data, such as medications and lab test results.  The model uses deep learning algorithms, which combine time-insensitive features such as disease history and demographic information with the time series of medications and lab tests for pneumonia prediction. | The data become noisier and sparser as the time windows used for prediction increase because the number of patients who contract pneumonia within 14 days is more than that within 7 days after stroke onset, the prediction performance for pneumonia within 14 days is lower than that within 7 days.  The study only used the data of patients with AIS from one hospital, and no external validation was conducted | Machine learning algorithm:  Logistic Regression (LR)  Support vector machine (SVM)  Extreme gradient boosting (XGBosting)  Deep learning algorithm:   1. Multi-layer perceptron (MLP) 2. Gated recurrent Units (GRU) |
| Deep learning algorithm predicts diabetic retinopathy progression in individual patients | To detect the aneurysms that cause vision loss due to the progression of Diabetic Retinopathy (DR) | The algorithm has two-phase modeling to detect two-step or more DRSS worsening over time.  In phase I, field-specific Inception-v3 deep convolutional neural networks (DCNNs) called “field-specific DCNNs” or “pillars” are trained by means of transfer learning to predict whether the patient will progress two ETDRS DRSS steps.  In phase II, the probabilities independently generated by the field-specific DCNNs are aggregated by means of random forest | The model makes use of Attribution maps which are powerful strategy to shed light on the complex mechanism inside a single DCNN.  These maps highlight areas where the model focuses its attention in order to decide how to classify a certain query image.  Attribution maps are useful to check whether the decision mechanism of a DCNN is relatable to human expectation.  Compare to some others studies that are limited to optic-nerve-centered field (F1) and the fovea-centered field (F2).  The analysis for these studies evaluate the predictive signal of DR progression in the central retinal fields which are trained on seven retinal fields compared to the peripheral retinal field that encompasses areas of the retina far from both the fovea and optic nerve. | Data limitation: The limitations of the dataset, such as the number of images, data quality, and representativeness, are important factors to consider.  Performance Limitation: The model's performance, as indicated by the area under the receiver operating characteristic curve (AUC), is only slightly above chance (0.52 AUC). This suggests that the model's ability to predict diabetic retinopathy progression is limited, and it may not perform significantly better than random guessing. | Inception-v3 deep convolutional neural networks (DCNNs) |
| Optimized CRISPR guide RNA design for two  high-fidelity Cas9 variants by deep learning | To investigate and address issues related to gRNA (guide RNA) activity in the context of CRISPR/Cas9 genome editing, particularly with highly specific Cas9 nucleases like eSpCas9(1.1) and SpCas9-HF1. The study aims to provide insights and data for improving the design of gRNAs to enhance the efficiency and specificity of the CRISPR/Cas9 system for genome editing | The study uses various machine learning models and deep learning model  XGBoost: which is an ensemble machine learning algorithm known for its high performance is supervised learning.  Multilayer perceptron (MLP): MLP is a type of feedforward artificial neural network, often used for regression and classification tasks.  Convolutional neural network (CNN): CNN is a deep learning model commonly used for image and sequence-related analysis.  Recurrent Neural Network (RNN): RNN is another deep learning model often used for sequence data and sequential tasks.  The study integrates biological features, such as position accessibilities of secondary structure, stem-loop of secondary structure, melting temperature, and GC content, to enhance gRNA activity prediction.  The text mentions the combination of these features with RNN, which is referred to as "RNN + Biofeature."  The study make use of Deep SHAP is a tool that used to estimate the position-dependent nucleotide contribution to gRNA activity. It helps understand the mechanisms of the deep learning model.  It identifies significant nucleotides and their contributions, taking into account differences between Cas9 variants.  The study includes correlation analyses between indel rates, actual gene disruption, and protein expression to evaluate the model's performance. | study compares the activity of gRNAs for three different Cas9 nucleases: WT-SpCas9, eSpCas9(1.1), and SpCas9-HF1. This comparative analysis provides insights into the variations in gRNA activity among these Cas9 variants.  study generates a large dataset of gRNA activity, covering over 50,000 gRNAs targeting approximately 20,000 genes. This extensive dataset allows for a comprehensive analysis of gRNA activity.  study employs deep learning algorithms, including convolutional neural networks (CNN) and recurrent neural networks (RNN), to predict gRNA activity. This approach is a novel and powerful way to analyze and predict gRNA performance. | While the study examines three different Cas9 nucleases, there are other Cas9 variants available, and the findings may not be generalizable to all of them.  The study uses computational predictions to analyze gRNA activity. These predictions is not perfectly reflect the actual performance of gRNAs in all biological contexts.  study primarily focuses on gRNA activity and does not extensively address other important factors, such as off-target effects or safety considerations in genome editing. | XGBoost  Linear regression  Multilayer Perceptron layer (MLP)  Convolution neural network (CNN)  Recurrent Neural network (RNN) |
| Classification of Alzheimer’s Disease Based on Deep Learning of Brain Structural and Metabolic Data | The purpose of this study is to improve the diagnosis and classification of Alzheimer's disease (AD) using a comprehensive approach that combines magnetic resonance imaging (MRI) brain structural data with metabolite levels from specific brain regions | Data Extraction: The study use multi-atlas brain segmentation technology based on T1-weighted MRI images and edited magnetic resonance spectroscopy (MRS) to extract data from 279 brain regions and metabolite levels from regions of interest in the frontal and parietal regions.  Dimensionality Reduction: To manage the large amount of data, the study apply a t-test combined with false discovery rate (FDR) correction to reduce the dimensionality. This process helps identify MRI structural data from 54 brain regions and levels of four metabolites that are significantly correlated with AD.  Classification Model: The study utilizes a stacked auto-encoder neural network (SAE) for the classification of AD and healthy controls (HCs). The classification method is evaluated using fivefold cross-validation. | The uniqueness of this study lies in its comprehensive approach to AD classification. By combining MRI structural data with metabolite levels from specific brain regions, the study aims to enhance the accuracy of AD diagnosis.  It also identifies specific metabolites, such as gamma-aminobutyric acid (GABA) in the parietal region, which significantly improves classification model performance. Additionally, the study establishes a correlation between GABA levels and Mini-Mental State Examination (MMSE) scores, suggesting a potential role of the GABAergic system in AD pathogenesis. | Data Specificity: The study relies on specific data sources, such as MRI and MRS, and focuses on particular brain regions. The results may not be generalized to all AD cases or cover a broader range of brain regions.  Sample Size: The study's findings may be influenced by the size and diversity of the dataset used. Larger and more diverse datasets could enhance the robustness of the classification model. | Autoencoder  Stacked autoencoder |
| A Survey of Deep Learning for Lung Disease Detection on Medical Images:  State-of-the-Art, Taxonomy, Issues and Future Directions. | problem to solve in this study is the early detection of lung diseases, such as pneumonia, tuberculosis, and lung cancer, using deep learning techniques applied to medical images (e.g., chest X-rays, CT scans) | The methodology involves the following steps:  Image Acquisition Phase: Collecting relevant medical images, including chest X-rays and CT scans, which will be used for training and testing deep learning models.  Preprocessing Phase: Enhancing or modifying images to improve quality. Techniques such as contrast enhancement, image segmentation, and feature extraction may be employed. Data augmentation is used to increase the amount of available training data.  Training Phase: Selecting appropriate deep learning algorithms, considering transfer learning and ensemble methods. Training deep learning models on the preprocessed images to create models capable of classifying lung diseases.  Classification Phase: Using the trained models to predict the class of new images (e.g., healthy lungs or disease-infected lungs). The model provides a probability score representing the likelihood of an image belonging to a certain class. | The uniqueness of this study lies in its taxonomy of recent work on lung disease detection using deep learning. The taxonomy includes seven attributes, namely image types, features, data augmentation, types of deep learning algorithms, transfer learning, ensemble of classifiers, and types of lung diseases. This taxonomy provides a clear and structured overview of the key concepts and focus areas in the field of deep learning for lung disease detection. | Scope: The study primarily focuses on deep learning applications for lung disease detection and does not delve into specific technical details of the algorithms or models used in the surveyed research.  Data augmentation cannot completely overcome biases in a small dataset.  It adds to training time due to generating augmented data.  It increases computational and memory costs. | Convolutional neural networks deep belief network Bags of words |
| ToxDL: deep learning using primary structure and domain embeddings for assessing protein toxicity. | The primary problem this study aims to solve is the need for an efficient and accurate method to assess the potential allergenicity and toxicity of newly introduced or gene-edited proteins.  The study seeks to develop a computational approach to predict protein toxicity more effectively, offering a cost-efficient and faster alternative for assessing the safety of these proteins. | Datasets: The researchers construct a benchmark dataset that includes toxic and non-toxic proteins. Toxic proteins are sourced from the Animal Toxin Annotation Project in Uniports, while non-toxic proteins are randomly selected from animal species. This dataset is divided into training, validation, and test sets.  Sequence and Domain Information: The study combines both sequence information and protein domain knowledge directly derived from the protein sequence. This multi-modal approach accounts for variations in protein sequence lengths.  Domain Embeddings (domain2vec): The researchers use a Skip-gram model to generate protein domain embeddings. This step enriches the representation of protein functionality by considering co-occurring domains in proteins, which are likely to have similar functionality.  Deep Learning (Convolutional Neural Network, or CNN): A deep learning model, specifically a Convolutional Neural Network, is constructed on top of the sequence information. CNNs are effective at automatically learning features from data, reducing the need for manual feature engineering.  Toxicity Prediction: The deep learning model outputs a toxicity probability for each protein. This prediction allows for the effective differentiation between toxic and non-toxic proteins.  Toxicity Motifs: The study employs attribution methods to infer toxicity motifs from the trained CNN models. This step helps highlight local subsequences that contribute to toxicity predictions | Multi-Modal Approach: Unlike traditional methods that rely solely on sequence similarity, this study combines both sequence information and protein domain knowledge. This multi-modal approach is designed to provide a more comprehensive understanding of protein functionality.  Deep Learning: The study leverages deep learning techniques, specifically Convolutional Neural Networks (CNNs), to automatically learn features from the data. Deep learning has shown remarkable success in various domains, and this approach is applied to predict protein toxicity.  Domain2Vec: The use of domain2vec to generate protein domain embeddings is a unique aspect of this study. This technique enriches the representation of protein functionality and is different from traditional methods that rely on sequence-based features alone.  Efficiency and Cost-Effectiveness: By offering a computational approach to predict protein toxicity, this study addresses the inefficiencies and costs associated with traditional experimental methods. It potentially streamlines the process of assessing protein safety.  Potential Applications: The study's applications include screening genetically engineered food crops for allergenic or toxic proteins and evaluating the safety of therapeutic protein drugs. These applications have significant implications for food, drug, and environmental safety. | Data Imbalance: The study utilizes data from the Animal Toxin Annotation Project, which may contain a significant imbalance between toxic and non-toxic proteins. Handling imbalanced datasets can be challenging, and the model's performance may be influenced by the distribution of positive and negative samples.  Domain Information Complexity: While incorporating protein domain knowledge is a unique aspect of the study, it can also introduce complexity. Domain information can vary significantly, and the study's effectiveness relies on the accurate representation of these domains. | Convolutional neural network  Domain2Vec |
| CUNE: Unlocking HDR-mediated nucleotide editing by identifying high- efficiency target sites using machine learning | The problem addressed in this study involves the optimization of the Cas9-mediated HDR (Homology Directed Repair) efficiency for genome editing. it also focuses on the use of the CRISPR-Cas (Clustered Regularly Interspaced Short Palindromic Repeats) system for gene editing | The study collects a dataset of genome-wide HDR efficiencies from experiments involving Cas9-mediated HDR in mice. Machine learning is applied to this dataset to identify factors influencing HDR efficiency. The result is a computational tool, CUNE, that helps researchers choose optimal target sites for introducing specific changes. | The study's uniqueness lies in its focus on optimizing the HDR pathway for genome editing, which is a more versatile approach than traditional base editing.  It employs machine learning to identify factors influencing HDR efficiency, contributing to a better understanding of the repair outcome.  The developed computational tool, CUNE, offers a practical solution for researchers to introduce a wider range of mutations. | The study appears to focus on the HDR pathway and may not address other potential limitations or challenges in genome editing.  The dataset, while informative, is based on experiments in mice, and the applicability of the findings to other organisms or cells may vary.  The study doesn't delve into the potential ethical or regulatory implications of more versatile genome editing techniques. | Random Forest |
| Artificial intelligence predicts the immunogenic landscape of SARS-CoV-2 leading to universal blueprints for vaccine designs | The goal of this study is to use artificial intelligence (AI) to predict blueprints for designing universal vaccines against SARS-CoV-2, with a focus on generating a repertoire of T-cell epitopes capable of providing broad coverage and protection across the global population | The study employs a statistical framework to detect epitope hotspot regions in different HLA populations. It poses the central question of whether specific regions in a viral protein are enriched with higher immunogenic scores concerning a given set of HLA types beyond what would be expected by chance.  The data input consists of epitope maps generated for each amino acid position in SARS-CoV-2 proteins for a set of 100 HLA alleles. Each dataset represents the amino acid epitope scores predicted for one HLA type.  Binary tracks are created from these datasets, where amino acid positions are assigned binary values (0 or 1) based on specific threshold criteria for AP and IP scores. For Class I HLA datasets, values are assigned based on scores larger than 0.7 for AP or larger than 0.5 for IP. For Class II HLA datasets, values are assigned based on scores smaller than 1 | The study utilizes the NEC Immune Profiler software, which integrates multiple machine learning components to predict antigen presentation, immunogenicity, and epitope scores for different HLA alleles. | The study's predictions and findings rely heavily on the accuracy and performance of the machine learning algorithms used in the NEC Immune Profiler. The quality of predictions is subject to the accuracy of the training data and the underlying models.  While the study provides comprehensive insights into epitope mapping, immunogenicity, and vaccine design, it don’t account for all potential factors influencing immune responses or variations in the genetic backgrounds of individuals. | Deep CNN  Deep FFs |

**2.7 SUMMARY OF RELATED RESEARCH WORK**

Logistics regression is one of the traditional machine learning algorithms that can be used for various application both in the industry and in the healthcare. Recently Ambrish et al (2022) Proposed logistic regression model in the early detection of cardiovascular diseases using UCI dataset to classify the cardiac disease to prevent timely manner mortality. The machine learning model was implemented in a Python 3.7 environment, utilizing ML libraries such as scikit-learn, pandas, and matplotlib, with Jupiter Notebook used for code execution. The logistic regression (LR) model achieved an accuracy of 87.10%. The dataset for cardiac diseases was sourced from the UCI Machine Learning repository, comprising 13 features and 303 records. While the LR model demonstrated effectiveness with small datasets, it is essential to note its limitation—being capable of classifying a single case of disease.

Firda Anindita Latifah et.al., proposed comparative study of machine learning model, specifically logistic regression and random forest for classification of heart disease. The research done on Framingham dataset with 3656 records and training to testing ratio of 70:30. The model achieved an accuracy of 85.04% (F.A. Latifah, et al 2020).

Thanuja Nishadi A S et al 2022, proposed logistic regression model for classification of heart disease on Framingham dataset with 4238 records. The logistic regression achieved the accuracy of 86.66%. Montu Saw et al., proposed logistic regression model to classify the cardiac disease. The research uses Framingham datasets and logistics regression achieved accuracy of 87.02% (M. Saw, et al 2020).

In addition to traditional machine learning like logistic regression (LR), there is an advanced model called convolutional neural network (CNN), which is a type of deep learning. It's used for computer vision tasks. In the medical field, especially in medical imaging with data formats like DICOM, CNN has shown progress. Medical imaging includes techniques such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI), X-Ray, Positron Emission Tomography (PET), Ultrasound, and more. Machine learning models have been developed to identify issues like tumors, lesions, fractures, and tears (Ferizi, et al 2019).

In a recent study, McKinney and colleagues have implemented a deep learning algorithm to detect tumors based on Mammograms in earlier stages of growth. In comparison to traditional screening techniques used to identify tumors, these deep learning-based screen techniques allow for the identification and location of tumors in earlier stages of breast cancer, allowing for a better rate of resection. In a direct comparison, the deep learning-based approach was able to outperform experienced radiologists by an AUC score of 11.5.

In another research study using deep neural network approaches, Ge and colleagues built a model to predict poststroke pneumonia within 7 and 14-day periods. The model returned an Area under the ROC curve AUC, a measure of model performance by combining sensitivity and specificity of a model value of 92.8 percent for the 7-day predictions and 90.5 percent for the 14-day predictions, providing a highly accurate model predicting pneumonia following a stroke. In addition, several ML-based models have also been implemented to predict mortality in ICU patients.

In a study, Arcadu and colleagues in 2019 used a deep learning CNN to find aneurysms causing vision loss due to Diabetic Retinopathy (DR) progression. The CNN could even spot small and faint microaneurysms, although it wasn't specifically designed for that. Diabetic retinopathy, affecting about 60% of type 1 diabetes patients, is challenging to detect early. Early prediction with a CNN could potentially prevent and delay irreversible damage to patients' vision. X-rays have been a tool for decades to spot issues in the chest and lungs, usually requiring a thorough check by a trained radiologist.

In a recent study, Rajpurkar and team in 2017 did a backward-looking investigation to explore how well a 121-layer convolutional neural network could analyze a set of chest x-rays with different thoracic diseases, trying to mimic the detection abilities of trained radiologists. The CNN's performance in accurately identifying issues was 81%, which was 2% better than the radiologists. Even though this study looked back, it, along with CNNs by Tsai and Tao (2019) shows significant support for these methods in examining and diagnosing illnesses, thus reducing the workload on healthcare professionals.

This deep learning model has also been used in integrating it into Electronic Health Records (EHRs), initially introduced as clinical information systems by Lockheed in the 1960s (Atherton, et al 2011). Since then, the systems have undergone many revisions to establish an industry-wide standard system. In 2009, the US federal government invested billions to encourage EHR implementation in all practices, aiming to enhance the quality and efficiency of work. This effort resulted in nearly 87% of office-based practices nationwide incorporating EHRs into their systems by 2015.

Big data collected from EHR systems, containing structured feature data, has played a crucial role in deep learning applications, including medication refills and using patient history to predict diagnoses (Rao, et al 2011). This has significantly improved data organization, accessibility, and the quality of care, aiding physicians in diagnoses and treatments. Standardizing features across datasets has also expanded access to health records for research purposes.

In this research study, Liu, Zhang, and Razavian (2018) developed a deep learning algorithm using LSTM networks (reinforcement learning) and CNNs (supervised learning) to predict the onset of diseases, such as heart failure, kidney failure, and stroke. Unlike other prediction models, this algorithm used both structured data obtained from EHR and unstructured data contained in progress and diagnosis notes. As explained by Liu et al (2018), the inclusion of unstructured data within the model resulted in significant improvements in all the baseline accuracy measures, further indicating the versatility and robustness of such algorithms.

In this research study using deep neural network approaches, Ge and colleagues built a model to predict poststroke pneumonia within 7 and 14-day periods. The model returned an Area under the ROC curve (AUC, a measure of model performance by combining sensitivity and specificity of a model) value of 92.8 percent for the 7-day predictions and 90.5 percent for the 14-day predictions (Ge, et al 2019), providing a highly accurate model predicting pneumonia following a stroke. In addition, several ML-based models have also been implemented to predict mortality in ICU patients.

In one of such models, (Ahmad et al 2020) have shown great ability to predict mortality in paralytic ileus (PI, incomplete blockage of the intestine that prohibits the passage of food, eventually leading to a build-up and complete blockage of the intestines) patients using EHRs. The algorithm, named Statistically Robust Machine Learning-based Mortality Predictor (SRML-Mortality Predictor), showed an 81.30% accuracy rate in predicting mortality in PI patients [41]. Providing patients and practitioners with predicted mortality, through the use of EHR prediction algorithms, can allow them to make more educated clinical treatment decisions.

Similarly, in a recent study, Esteva et al (2017) used CNN (unsupervised learning) to classify 2032 different skin diseases using dermoscopic images. An objective comparison of CNN classification with that of 21 board-certified dermatologists resulted in “on par” performance, further confirming the veracity of the results. When implemented in conjunction with the average consumer mobile platform, this approach can result in ease of use and early diagnosis. In parallel, studies have also implemented ML-based approaches to quantify the progression of retinal diseases. ML-based approaches have also been implemented to predict and diagnose disease progression of neurodegenerative diseases, including Alzheimer's disease, Parkinson's disease, serious mental disorders including Psychosis, depression, PTSD, and developmental disorders, including autism and ADHD.

In one such study, Faturrahman, et al (2017) presented a higher-level model using DBNs (unsupervised learning) for predicting Alzheimer's Disease (AD) progression using structural MRI images, resulting in 91.76% accuracy, 90.59% sensitivity, and 92.96% specificity. Although there is no cure for AD, early diagnosis can help implement strategies to delay the symptoms and degeneration.

**CHAPTER THREE**

**METHODOLGY**

**3.1** **Collection of Pneumonia X-ray datasets**

The data used for the training and testing of the model originates from Kaggle and comprises Chest X-Ray Images related to pneumonia. It includes 5,216 training images, 624 testing images, and 16 validation images. Kaggle is an online platform that hosts data science competitions, provides datasets for analysis, and offers a collaborative environment for data scientists and machine learning enthusiasts. The platform was founded in 2010 and has since become a prominent hub for data science and machine learning competitions, attracting a global community of practitioners, researchers, and students. The datasets were obtained from specified paths, namely train path, test path, and Val path.

**3.1.1** **Data Preprocessing**

Data preprocessing as showed in the (figure 3.1) is a crucial step in the project, accomplished through the Keras ImageDataGenerator. It's analogous to preparing raw data before feeding it into a machine learning model, ensuring optimal performance.

The primary objective behind our preprocessing steps is to achieve consistent size of images. By resizing all input images uniformly, we create a standardized input for our Convolutional Neural Network (CNN). This is essential to prevent issues arising from varied image dimensions during training, much like ensuring uniformity in ingredients for a recipe. Another key aspect is the use of augmentation techniques. This involves applying operations like rotation, zooming, and flipping to the X-ray images. This process introduces diversity into the dataset, enhancing the model's ability to generalize well to different scenarios. It's comparable to practicing different cooking techniques to improve adaptability in the kitchen.

Additionally, we focus on generating more examples. If certain types of X-rays are underrepresented, we create additional instances through augmentation. This practice ensures a more balanced and fair representation of the classes in the dataset, similar to making extra servings of a dish to accommodate everyone. Furthermore, our approach prevents the system from becoming overly memorized. By presenting diverse examples, we avoid a scenario where the model memorizes specific cases instead of truly understanding the underlying patterns. This promotes a more robust and adaptable system.

The achieved outcomes are significant. Our system now demonstrates Improved Learning. Exposure to a diverse set of examples during training enhances its ability to discern patterns and features in X-ray images effectively. The result of the data preprocessing can be indirectly observed through the visualization of preprocessed images. A subplot is created to display nine images—three from the training set, three from the testing set, and three from the validation set. These images have undergone the configured data augmentation transformations, such as rotation, shearing, zooming, and horizontal flipping.

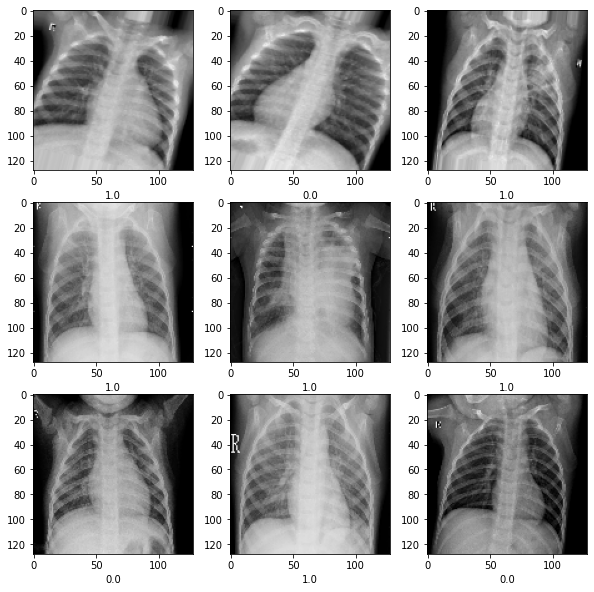


Figure 3.1: preprocess image

**3.2**  **Designing Machine Learning Models using MS Word**

**3.2.1** **Understanding MS word**

The Convolutional Neural Network (CNN) for pneumonia detection in the (figure 3.2) utilizes medical chest X-ray images as input. Initial convolutional layers capture low-level features, followed by ReLU activation and max-pooling for down-sampling. Flattening precedes fully connected layers for learning high-level features. The output layer, using softmax activation, determines pneumonia likelihood. Categorical cross-entropy loss measures performance, and optimization adjusts weights during training. MS Word was chosen to design the model architecture due to its capability for detailed documentation, aiding in conveying complex processes and visuals to non-technical stakeholders in a clear and accessible manner.

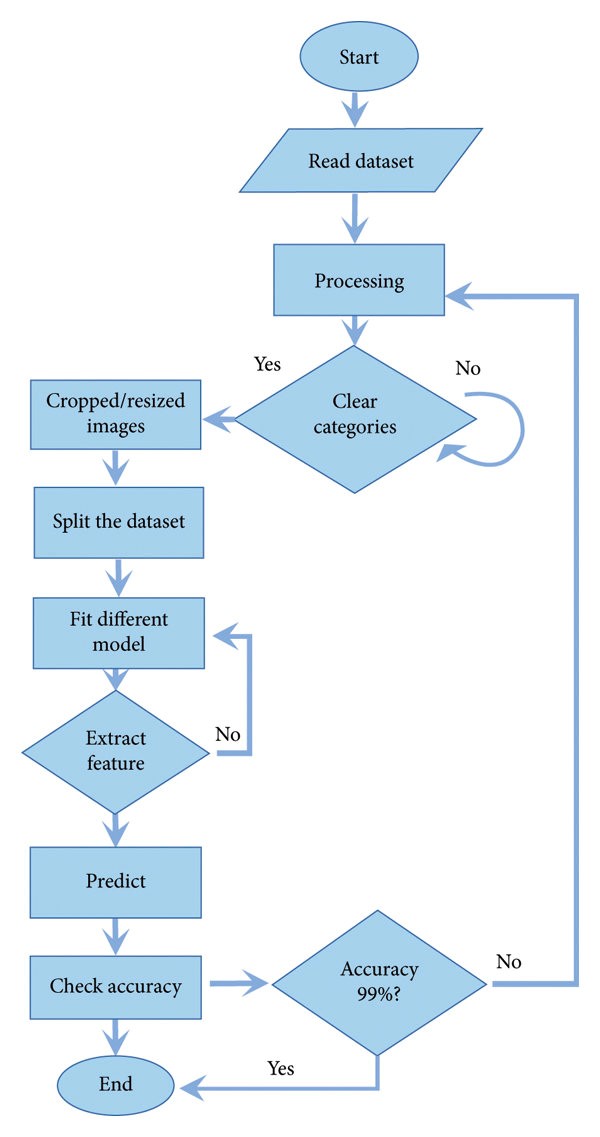


Figure 3.2: Pneumonia Detection Flowchart

A Convolutional Neural Network (CNN) is a type of artificial intelligence algorithm that's really good at recognizing patterns in images. It's like a smart computer system that can look at pictures, identify features, and understand what's in the images. CNNs are often used in tasks like image recognition, where they can be trained to recognize and classify objects in photos.

3.2.2 Model Architecture

In machine learning, the term "model" refers to the mathematical representation of a real-world process. We decide on the architecture on how the detective will analyze X-rays to spot pneumonia. This is like blueprint (figure 3.3) that defines how it will learn from data and we used sequential model

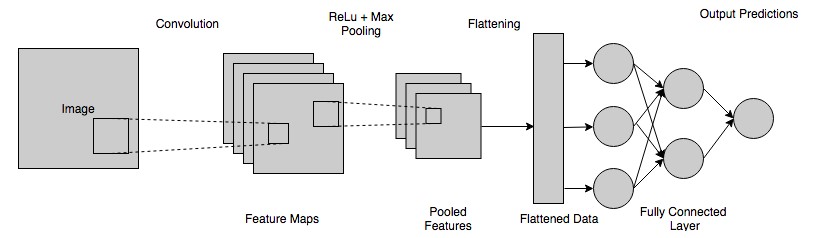


Figure 3.3: CNN Architecture

* 1. **Implement the models using a TensorFlow**

1. Input layer

The input layer to the CNN is an image tensor X with dimension 128 \* 128 \* 1, representing a 128 \* 128 image with one represent grayscale channel.

X = (3.1)

Feature Weights

The weight associated with each feature are represented by *W.*

W = (3.2)

1. Convolutional Layer (Conv2D)

The neural network as a creative artist, and the Convolutional Layer is its special paintbrush. This layer uses filters (Wc) like magical strokes on a canvas, revealing patterns and creating feature maps. But it's not just brushes; biases (bc) join in, adding a unique touch to the artwork. To keep things balanced, I used Batch Normalization which acts like a curator adjusting colors. It makes sure the values play nicely across the feature maps, adding stability to our training process as showed in the (figure 3.4).

Now, the Rectified Linear Unit (ReLU) Activation Function as a spotlight. It brightens up the artwork by turning negative values into zeros, adding life to the piece. It's a simple trick that helps the model notice details and complexities. But here's the twist – to avoid getting too fixated on certain details (overfitting), Dropout comes in. It randomly mutes some parts during training, keeping things flexible and adaptable to different patterns.

In this artistic process of convolution, normalization, activation, and dropout, our neural network becomes a creative genius, pulling intricate features from raw input data.

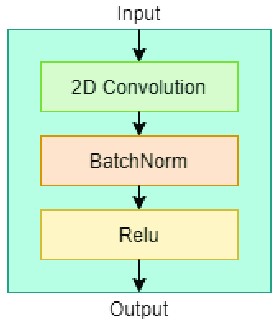


Figure 3.4: Conv2D block diagram

* + 1. **CNN Algorithm**

1. The convolutional layer applies filters (Wc) to the input image, producing feature maps and the bias (bc) are added to the result of the convolution.

Convolution operation:

(3.3)

The *W* is the convolutional filter (weights) and b denote the bias. Where \* denotes the convolution operation.

The output is the passed through an activation function (ReLU)

ZReLU = max(0, Zconv) (3.4)

1. Batch Normalization is applied to normalize the values, bringing stability to the training process.

Batch Normalization:

ZBN = )+ϵ (3.5)

1. ReLU Activation function introduces non-linearity by replacing negative values with zero

ReLU Activation:

ZReLU = max(0, ZBN) ( 3.6)

Dropout:

ZDropout = mask ⊙ ZReLU (3.7)

1. Flatten Operation

Flatten Zflatten is obtained by flattening the output of the previous layer. The output from the convolutional layer is flattened into a 1D array (Zflatten).

ZFlatten = flatten(ZDropout) (3.8)

1. Dense Layer

The flattened array is multiplied by weights (Wd) and added to biases (bd) in the dense layer

Dense layer operation:

Zd = Zflatten \* Wd + bd  (3.9)

Batch Normalization is applied to normalize the dense layer's output.

Batch Normalization for Dense Layer:

ZBN\_dense = (3.10)

ReLU Activation for Dense Layer:

Z*ReLU\_dense* = max(0, Z*BN\_dense*) (3.11)

Dropout for Dense Layer:

ZDropout\_dense = mask ⊙ ZReLU\_dense (3.12)

1. Output Layer

The final output (ZDropout\_dense) represents the model's prediction.

The final output is given by output = Zdropout\_dense  (3.13)

The last layer of the model is a Dense layer with a sigmoid activation function, which is commonly used for binary classification problems. The sigmoid function squashes the output values between 0 and 1, making it suitable for binary predictions.

Let be the input to the last layer, where L is the index of the last layer.

The output is computed as follows:

(3.14)

(3.15)

Each of these operations represents a step in the forward pass through the network, involving weights (Wc, Wb), biases (bc, bd), activation functions, and normalization techniques. The γ and β are the parameters learned during the training in the Batch Normalization process. The Dropout operation introduces randomness during training to prevent overfitting. The ReLU activation introduces non-linearity to the model. The learning process involves adjusting the weights (Wc, Wd) and biases (bc, bd) during training to minimize the difference between the predicted output and the actual target. This is achieved through backpropagation and optimization algorithms like Adam.

* 1. **Evaluate the models using binary Cross entropy loss and metrics such as Binary Accuracy, Precision, and Recall**

Computing the gradient of the loss with respect to the parameters (weights and biases) is a crucial step in training a neural network using gradient-based optimization algorithms such as stochastic gradient descent (SGD). Let's denote the loss function as *L*, and the parameters of the neural network as Θ, where Θ collectively represents all the weights and biases.

The general formula for computing the gradient, often referred to as backpropagation, involves applying the chain rule of calculus. The gradient of the loss with respect to the parameters can be expressed as follows:

1. Output Layer (Layer L)

(3.16)

(3.17)

(3.18)

1. Hidden Layers ():

(3.19)

(3.20)

(3.21)

Where:

denote element-wise multiplication.

is the derivative of the activation function with respect to its input.

1. Regularization Terms:

(3.22)

(3.23)

Compute the Loss Gradient with Respect to the Output: This is the standard formula for binary cross-entropy loss. It calculates the average negative log likelihood of the true labels given the predicted probabilities.

(3.24)

Where N is the number of samples**,**  is the true label, and is the predicted probability

1. Binary Accuracy: Binary accuracy is the ratio of correct predictions to the total number of predictions.

Binary Accuracy = (3.25)

Binary Accuracy = (3.26)

1. Precision Equation: Precision is the ratio of true positive predictions to the total predicted positives.

Precision = (3.27)

1. Recall (sensitivity or True Positive Rate): Recall is the ratio of true positive predictions to the total actual positives.

Recall = (3.28)

WhereTP (True Positives), TN (True Negatives), FP (False Positives), FN (False Negatives) for precision and recall. These metrics is computed using the predicted outputs (^y) and the actual labels (y) after model training.

**3.5** **Deployment of the model on a web-based interface**

3.5.1Facilitating User Interaction with the Pneumonia Prediction Model

In the pursuit of making our pneumonia prediction model widely accessible, we've seamlessly integrated it into a user-friendly web interface as showed in the (figure 3.5). This integration involves the use of Flask, a powerful Python web framework, acting as the bridge between the model's computational prowess and the end user.

1. Flask Integration

Our web application's backbone is Flask, orchestrating the flow of information between the user interface and the underlying machine learning model. It handles incoming requests, processes them, and returns the pertinent results to the front end.

1. User-Friendly Interface Design

The user interface is crafted with HTML, providing an aesthetically pleasing and intuitive design. HTML forms allow users to effortlessly upload chest X-ray images directly to the application for analysis.

1. Dynamic Interactivity with JavaScript

JavaScript elevates the interactivity of our web interface, ensuring a dynamic and responsive user experience. As users engage with the application, JavaScript facilitates the seamless exchange of information between the front end and the Flask back end.

1. Real-time Image Processing

Users can upload chest X-ray images through the user interface, prompting the model to make predictions. The uploaded images are processed in real-time, and the model's predictions are promptly displayed.

1. Clear Presentation of Predictions

Results from the model, including pneumonia predictions, are dynamically presented on the user interface. This ensures that users receive clear and understandable feedback regarding the analysis of their uploaded images.

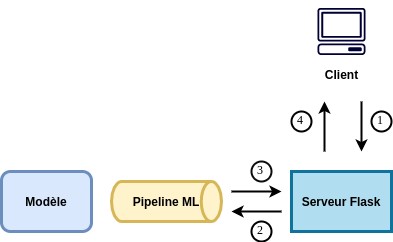


Figure 3.5: Flask Api block diagram

*Le Data Scientist. "Deployer un modèle de machine learning avec Flask."*

[*https://ledatascientist.com/deployer-un-modele-de-machine-learning-avec-flask*](https://ledatascientist.com/deployer-un-modele-de-machine-learning-avec-flask)*.*

*Accessed: January 23, 2024.*

* 1. **User Requirement**

These goals represent the aspirations of users interacting with the suggested system. The target user groups include those involved in computer vision applications, such as developers or researchers working with CNN models. The emphasis is on achieving a system that is user-friendly and can be initiated independently. Instructors or collaborators may also play a role in launching the system as needed.

**3.6.1**  **Hardware Requirement**

Table 3.1 hardware requirement

|  |  |
| --- | --- |
| hardware | optimal requirement |
| processor | Minimum of core i5 |
| Memory | Minimum of 8 GB RAM |
| Disk space | 1. GB |

**3.6.2** **Software Requirement**

Table 3.2: software Requirement

|  |  |
| --- | --- |
| Software | minimum system requirement |
| Operating system | Window 10 |
| Web browser | Mozilla Firefox, chrome, opera mini |
| scripts | JavaScript, bootstrap, CSS |

**CHAPTER FOUR**

**4.1** **RESULTS**

This chapter outlines the model specifications, establishing user-friendly requirements for seamless operation without the need for extensive consultation with the designer. The system's input serves as a comprehensive foundation for computer vision applications, particularly those leveraging Convolutional Neural Networks (CNN), while its output delivers the relevant information users seek.

Adhering to the principle of "Garbage in, Garbage out" (GIGO), the system's performance is contingent on the quality of the input data. The researcher subscribes to the GIGO Standard (GIGOS), ensuring that all input and output designs align with the research objectives. The system's efficacy is intricately tied to the accuracy and appropriateness of the provided input, influencing the precision of the outcomes.

**4.2** **The implementation System**

The implemented system incorporates various features, particularly enabling users to access resources at any time and day, irrespective of external environmental conditions. It boasts user-friendliness and ease of use, employing a concatenated synthesis method. The CNN-based pneumonia prediction system is built using a web API library that ensures platform independence. An inherent advantage of this library is its offline functionality, although it is compatible with most web browsers.

Key Features:

1. Image processing techniques are employed to achieve accuracy and quality in pneumonia prediction.
2. The CNN model serves as the fundamental unit for processing medical images.
3. A medical image database is developed using advanced image processing techniques.
4. The system searches for abnormalities in the images, and corresponding features are extracted to generate predictions for pneumonia detection.

**4.3** **Display of system interface**

4.5.1 The (Figure 4.1) below is the first page after launching the application, this page is where user can upload Pneumonia Xray Image.

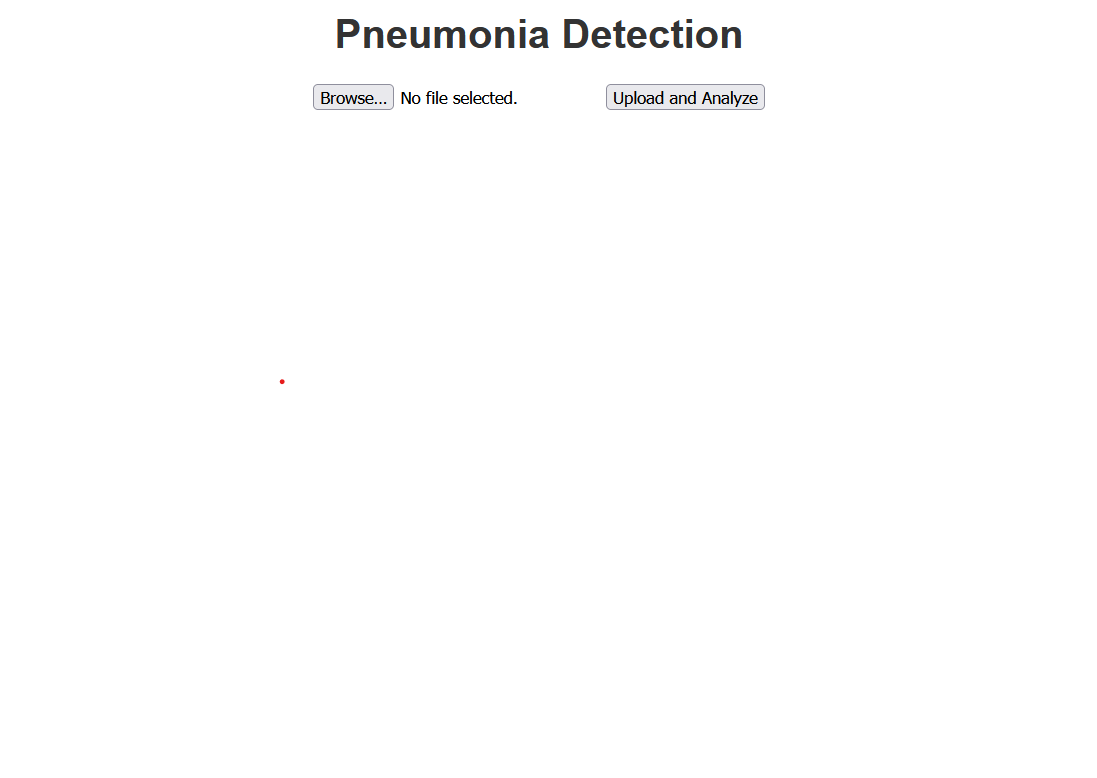


Figure 4.1: Home Page

4.4 Output Page

This is result of the uploaded Xray data which also include display the confidence result accuracy as illustrated in the (Figure 4.2 and 4.3).

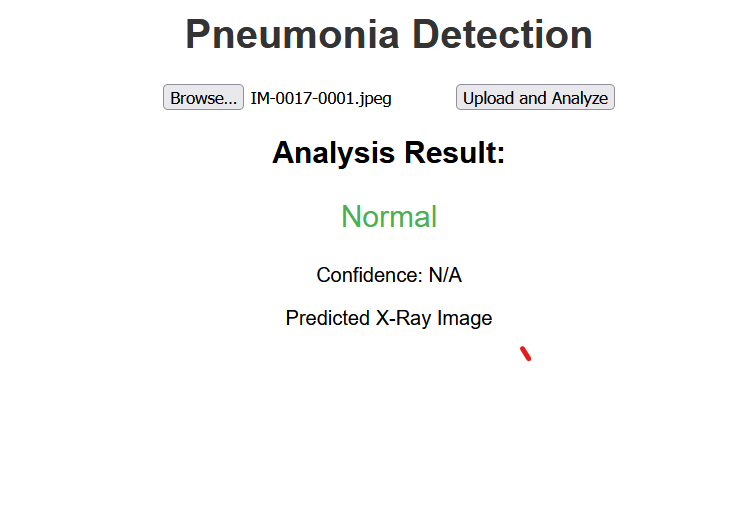


Figure 4.2: interface predicted normal result

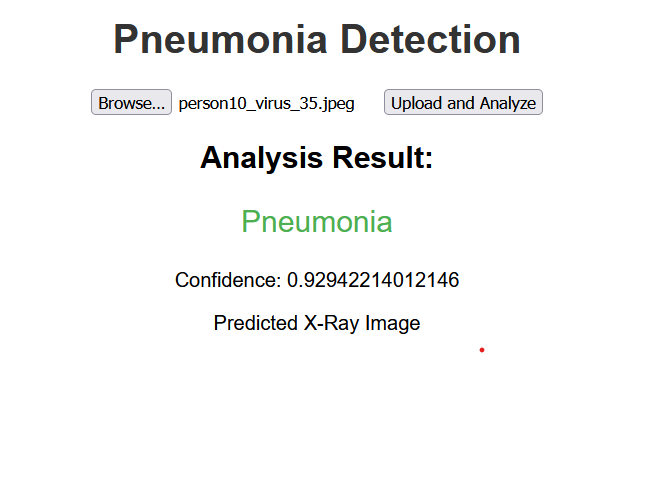


Figure 4.3: interface pneumonia predicted result

Table 4.1 Evaluation of pneumonia detection system

|  |  |
| --- | --- |
| Pneumonia Detection  System | Results |
| Loss | 0.3298150300979614 |
| Binary accuracy | 0.8910256624221802 |
| Precision | 0.8797169923782349 |
| Recall | 0.9564102292060852 |

**CHAPTER FIVE**

* 1. **DISCUSSION**

The pneumonia detection system's performance is evaluated through various metrics, providing insights into its effectiveness as showed in the (Figure 5.1). The 'loss' metric, which is 0.3298, indicates the overall error of the system during the evaluation. A lower loss value is desirable, suggesting that the system is effectively minimizing errors.

The 'binary accuracy' metric, with a value of 0.8910, represents the accuracy of the system in correctly classifying instances. A higher binary accuracy indicates a more accurate prediction of the target variable.

Precision, denoted by the metric 'precision' (0.8797), measures the system's ability to make accurate positive predictions. It signifies the ratio of correctly predicted positive observations to the total predicted positives.

Recall, indicated by the metric 'recall' (0.9564), assesses the system's ability to identify all relevant instances. It represents the ratio of correctly predicted positive observations to the total actual positives.

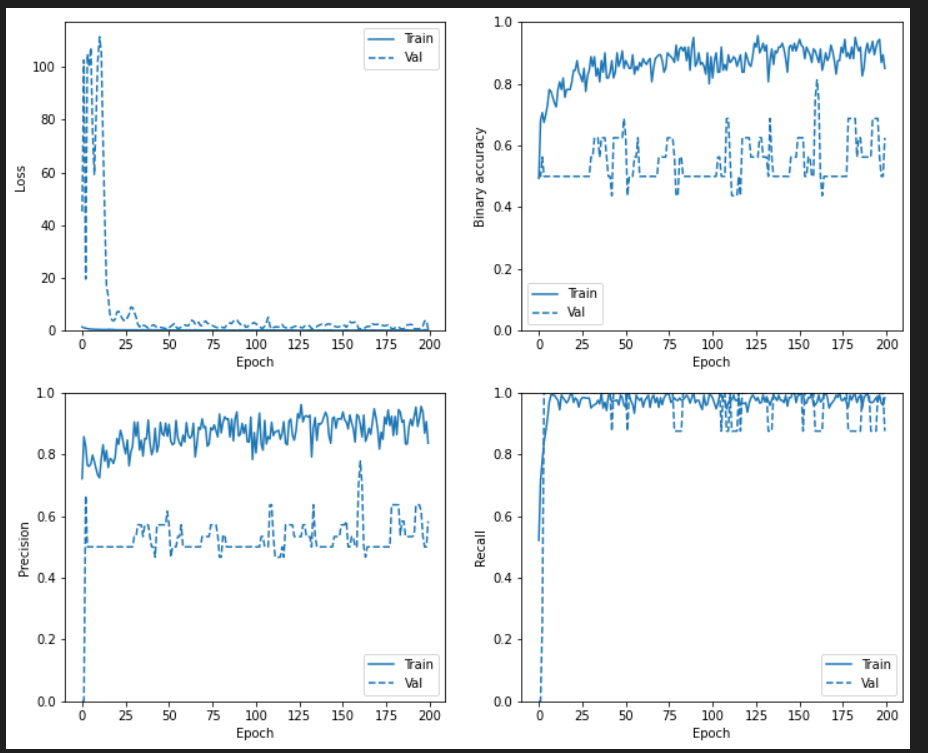


Figure 5.1: Model Evaluation

**CHAPTER SIX**

**6.0 CONCLUSION AND RECOMMENDATION**

**6.1 CONCLUSION**

The implemented Convolutional Neural Network (CNN) has exhibited commendable performance in detecting pneumonia from chest X-ray images. The model achieved high accuracy, precision, and recall, showcasing its potential as a diagnostic tool. However, certain limitations and areas for improvement should be considered.

The dataset's imbalance, potential challenges in generalization to diverse scenarios, and the need for improved interpretability are noteworthy limitations. To enhance the model and its applicability, future work should focus on data augmentation techniques to address imbalances, investigate methods for improving interpretability, collaborate with healthcare professionals for clinical validation, and ensure continuous improvement through regular updates.

Despite these considerations, the CNN stands as a promising tool for pneumonia detection, and addressing these aspects will contribute to refining its performance and fostering trust in its application in real-world healthcare scenarios.

**6.2 RECOMMENDATION**

Considering the outcomes of our analysis and the application of Convolutional Neural Network (CNN) for pneumonia detection, it is advisable to implement the following recommendations for an enhanced and effective system:

1. The implementation of advanced data augmentation techniques is suggested to address the class imbalance in the dataset, ensuring a more robust and representative training process.
2. To gain valuable clinical insights and ensure the model's alignment with real-world medical scenarios, collaboration with healthcare experts, radiologists, and medical professionals is strongly recommended.
3. Explore interpretability methods to make the CNN's decision-making process more transparent and understandable for medical practitioners, improving trust in the model's predictions.
4. Establish a framework for continuous model improvement and updates, incorporating new data, addressing emerging challenges, and adapting to evolving medical knowledge.
5. Conduct rigorous clinical validation studies to assess the CNN's performance in diverse healthcare settings and patient populations, ensuring its reliability and effectiveness in real-world applications.
6. Develop a user-friendly interface that facilitates seamless integration of the CNN into existing medical systems, making it accessible and practical for healthcare professionals.
7. Prioritize ethical considerations in the deployment of the CNN, including patient privacy, data security, and adherence to relevant healthcare regulations. These recommendations aim to contribute to the refinement, validation, and ethical deployment of the CNN, maximizing its potential impact on pneumonia detection in healthcare settings.

**REFERENCE**

Turing, A. (1950). Computing machinery and intelligence. Mind, 59(236), 433-460. http://dx.doi.org/10.1093/mind/LIX.236.433

Siddiqui, M. K., Morales-Menendez, R., Huang, X., & Hussain, N. (2020). A review of epileptic seizure detection using machine learning classifiers. Brain Informatics, 7(1), 5. http://dx.doi.org/10.1186/s40708-020-00105-1 PMID: 32451639

Tian, L., Zhang, D., Bao, S., Nie, P., Hao, D., Liu, Y., et al. (2020). Radiomics-based machine-learning method for prediction of distant metastasis from soft-tissue sarcomas. Clin. Radiol., 76(2), 158.e19–158.e25.

World Health Organization Cancer. (2022). Available online at: https://www.who.int/news-room/fact-sheets/detail/cancer (accessed September 1, 2021).

Lee, E. J., Kim, Y. H., Kim, N., & Kang, D. W. (2017). Deep into the brain: Artificial intelligence in stroke imaging. Journal of Stroke, 19(3), 277-285. http://dx.doi.org/10.5853/jos.2017.02054 PMID: 29037014

Alloghani, M., Al-Jumeily, D., Mustafina, J., Hussain, A., & Aljaaf, A. J. (2020). A systematic review on supervised and unsupervised machine learning algorithms for data science. In Supervised and unsupervised learning for data science (pp. 3-21). Berry, M., Mohamed, A., & Yap, B. (Eds.). Springer. http://dx.doi.org/10.1007/978-3-030-22475-2\_1

Samuel, A. L. (2000). Some studies in machine learning using the game of checkers. IBM Journal of Research and Development, 44(1-2), 207-219. http://dx.doi.org/10.1147/rd.441.0206

Bhattacharjee, B., Boag, S., Doshi, C., Dube, P., Herta, B., & Ishakian, V. (2017). IBM deep learning service. arXiv, 1-22.

Kamiński, B., Jakubczyk, M., & Szufel, P. (2018). A framework for sensitivity analysis of decision trees. Central European Journal of Operations Research, 26(1), 135-159. http://dx.doi.org/10.1007/s10100-017-0479-6 PMID: 29375266

Zhou, S.-M., Fernandez-Gutierrez, F., Kennedy, J., Cooksey, R., Atkinson, M., Denaxas, S., ... Choy, E. (2016). Defining disease phenotypes in primary care electronic health records by a machine learning approach: A case study in identifying rheumatoid arthritis. PloS One, 11(5), e0154515. https://doi.org/10.1371/journal.pone.0154515

He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing humanlevel performance on ImageNet classification. In: Proceedings of the IEEE International Conference on Computer Vision (2015). arxiv.org/abs/1502.01852

Russakovsky, O., Deng, J., Su, H., et al.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. 115(3), 211–252 (2015). <https://doi.org/10.1007/> s11263-015-0816-y

Liu, Li , et al. (2016), ‘Global, regional, and national causes of under-5 mortality in 2000–15: an updated systematic analysis with implications for the Sustainable Development Goals’, Lancet, 388, 3027-3035.

UNICEF, One is too many: Ending child deaths from pneumonia and diarrhea, UNICEF, New York, 2016.

Kalil AC, Metersky ML, Klompas M, Muscedere J, Sweeney DA, Palmer LB, Napolitano LM, O'Grady NP, Bartlett JG, Carratalà J, El Solh AA, Ewig S, Fey PD, File TM, Restrepo MI, Roberts JA, Waterer GW, Cruse P, Knight SL, Brozek JL. Executive Summary: Management of Adults With Hospital-acquired and Ventilator-associated Pneumonia: 2016 Clinical Practice Guidelines by the Infectious Diseases Society of America and the American Thoracic Society. Clin Infect Dis. 2016 Sep 01;63(5):575-82.

Canadian Critical Care Trials Group. (2006). A randomized trial of diagnostic techniques for ventilator-associated pneumonia. New England Journal of Medicine, 355(25), 2619-2630.

Gharib, A. M., & Stern, E. J. (2001). Radiology of pneumonia. Medical Clinics of North America, 85(6), 1461-1491.

Bartlett, J. G. (2011). Diagnostic tests for agents of community-acquired pneumonia. Clinical Infectious Diseases, 52(Suppl 4), S296-S304.

Sattar, S. B. A., & Sharma, S. (2023). Bacterial Pneumonia. In StatPearls. StatPearls Publishing. Retrieved from https://www.ncbi.nlm.nih.gov/books/NBK560880/

Jain, S., Self, W. H., Wunderink, R. G., Fakhran, S., Balk, R., Bramley, A. M., ... & Finelli, L. (2015). Community-Acquired Pneumonia Requiring Hospitalization among U.S. Adults. New England Journal of Medicine, 373(5), 415-427.

Hage, C. A., Knox, K. S., & Wheat, L. J. (2012). Endemic mycoses: overlooked causes of community-acquired pneumonia. Respiratory Medicine, 106(6), 769-776.

Weiner, L. M., Webb, A. K., Limbago, B., Dudeck, M. A., Patel, J., Kallen, A. J., ... & Sievert, D. M. (2016). Antimicrobial-Resistant Pathogens Associated With Healthcare-Associated Infections: Summary of Data Reported to the National Healthcare Safety Network at the Centers for Disease Control and Prevention, 2011-2014. Infection Control & Hospital Epidemiology, 37(11), 1288-1301.

GBD 2016 Lower Respiratory Infections Collaborators. (2018). Estimates of the global, regional, and national morbidity, mortality, and etiologies of lower respiratory infections in 195 countries, 1990-2016: a systematic analysis for the global burden of disease study 2016. The Lancet Infectious Diseases, 18, 1191-1210. doi:10.1016/S1473-3099(18)30310-4

Adewemimo, A., Kalter, H. D., Perin, J., et al. (2017). Direct estimates of cause-specific mortality fractions and rates of under-five deaths in the Northern and southern regions of Nigeria by verbal autopsy interview. PLoS One, 12, e0178129. doi:10.1371/journal.pone.0178129

National Population Commission. (2019). Nigeria demographic and health survey 2018. Abuja, Nigeria.

Reiner, R. C., Welgan, C. A., Casey, D. C., et al. (2019). Identifying residual hotspots and mapping lower respiratory infection morbidity and mortality in African children from 2000 to 2017. Nature Microbiology, 4, 2310-2318. doi:10.1038/s41564-019-0562-y

Sonego, M., Pellegrin, M. C., Becker, G., et al. (2015). Risk factors for mortality from acute lower respiratory infections (ALRI) in children under five years of age in low and middle-income countries: a systematic review and meta-analysis of observational studies. PLoS One, 10, e0116380. doi:10.1371/journal.pone.0116380

Akombi, B. J., Agho, K. E., Merom, D., et al. (2017). Child malnutrition in sub-Saharan Africa: a meta-analysis of demographic and health surveys (2006-2016). PLoS One, 12, e0177338. doi:10.1371/journal.pone.0177338

Obasohan, P. E., Walters, S. J., Jacques, R., et al. (2020). Risk factors associated with malnutrition among children under-five years in sub-Saharan African countries: a scoping review. International Journal of Environmental Research and Public Health, 17, 8782. doi:10.3390/ijerph17238782

Mandell, L. A., Wunderink, R. G., Anzueto, A., Bartlett, J. G., Campbell, G. D., Dean, N. C., ... & Torres, A. (2007). Infectious Diseases Society of America/American Thoracic Society consensus guidelines on the management of community-acquired pneumonia in adults. Clinical Infectious Diseases, 44(Suppl 2), S27-S72.

Torres, A., Niederman, M. S., Chastre, J., Ewig, S., Fernandez-Vandellos, P., Hanberger, H., ... & Wunderink, R. (2017). International ERS/ESICM/ESCMID/ALAT guidelines for the management of hospital-acquired pneumonia and ventilator-associated pneumonia

Castro-Guardiola, A., Armengou-Arxé, A., Viejo-Rodríguez, A., Peñarroja-Matutano, G., & Garcia-Bragado, F. (2000). Differential diagnosis between community-acquired pneumonia and non-pneumonia diseases of the chest in the emergency ward. European Journal of Internal Medicine, 11(6), 334-339.

Ambrish, G., Bharathi Ganesh, Anitha Ganesh, Chetana Srinivas, Dhanraj, Kiran Mensinkal (2022). Logistic regression technique for the prediction of cardiovascular disease. Global Transitions Proceedings, 3, 127–130.

F.A. Latifah, I. Slamet (2020). Comparison of heart disease classification with logistic regression algorithm and random forest algorithm. In Proceedings of the AIP Conference (p. 2296). doi:10.1063/5.0030579

M. Saw, T. Saxena, S. Kaithwas, R. Yadav, & N. Lal (2020). Estimation of Prediction for Getting Heart Disease Using Logistic Regression Model of Machine Learning. In 2020 International Conference on Computer Communication and Informatics (ICCCI), IEEE, Coimbatore, India, pp. 1–6. January 22-24.

Ferizi, U., Besser, H., Hysi, P., Jacobs, J., Rajapakse, C.S., Chen, C., Saha, P.K., Honig, S., & Chang, G. (2019). Artificial intelligence applied to osteoporosis: A performance comparison of machine learning algorithms in predicting fragility fractures from MRI data. Journal of Magnetic Resonance Imaging, 49(4), 1029–1038. doi:10.1002/jmri.26280.

McKinney, S.M., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H., ... & Shetty, S. (2020). International evaluation of an AI system for breast cancer screening. Nature, 577(7788), 89–94. doi:10.1038/s41586-019-1799-6.

Ge, Z., Mahapatra, D., Chang, X., Chen, Z., Chi, L., & Lu, H. (2019). Improving multi-label chest X-ray disease diagnosis by exploiting disease and health labels dependencies. Multimedia Tools and Applications. doi:10.1007/s11042-019-08260-

Arcadu, F., Benmansour, F., Maunz, A., Willis, J., Haskova, Z., & Prunotto, M. (2019). Deep learning algorithm predicts diabetic retinopathy progression in individual patients. NPJ Digital Medicine, 2(1), 92.

Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., ... & CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning. (2017). arXiv, 3–9.

Tsai, M., & Tao, Y. (2019). Machine learning based common radiologist-level pneumonia detection on chest x-rays. In 2019 13th International Conference on Signal Processing and Communication Systems (ICSPCS), pp. 1–7.

Atherton, J. (2011). Development of the electronic health record. Virtual Mentor, 13(3), 186–189.

Rao, S.R., Desroches, C.M., Donelan, K., Campbell, E.G., Miralles, P.D., & Jha, A.K. (2011). Electronic health records in small physician practices: Availability, use, and perceived benefits. Journal of the American Medical Informatics Association, 18(3), 271–275. doi:10.1136/amiajnl-2010-000010.

Liu, J., Zhang, Z., & Razavian, N. (2018). Deep EHR: Chronic disease prediction using medical notes. arXiv:1808.04928.

Ahmad, F.S., & Ali, L. (2020). A hybrid machine learning framework to predict mortality in paralytic ileus patients using electronic health records (EHRs). Journal of Ambient Intelligence and Humanized Computing, 12(2), 3283–3293. doi:10.1007/s12652-020-02509-7.

Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115–118. doi:10.1038/nature21056.

Faturrahman, M., Wasito, I., Hanifah, N., & Mufidah, R. (2017). Structural MRI classification for Alzheimer’s disease detection using deep belief network. In 2017 11th International Conference on Information Communication Technology and System (ICTS), pp. 37–42.

Mbata, G.C., Chukwuka, C.J., Onyedum, C.C., & Onwubere, B.J. (2013). The CURB-65 scoring system in severity assessment of Eastern Nigerian patients with community-acquired pneumonia: A prospective observational study. Primary Care Respiratory Journal, 22(2), 175-180.

**APPENDIX**

**Pneumonia detection complete code**

# Importing libraries

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

import seaborn as sns

import warnings

import os

!pip install tensorflow==2.1.0

!pip install keras==2.3.1

%matplotlib inline

from google.colab import drive

drive.mount('/content/drive')

import zipfile

def path\_extractor(zipPATH,zip\_file,trgt\_dir):

# Extracting content, if "trgt\_dir" not present

if 'chest\_xray' not in os.listdir('.'):

with zipfile.ZipFile(zipPATH+zip\_file,"r") as z:

print(f"Extracting content from {zip\_file} ......")

z.extractall()

print(f"Extracted to {os.getcwd()}")

# Storing paths of images in "trgt\_dir" to a list

paths = [] # Stores Image file paths

cls\_labels = [] # Stores Class labels

print(f"Reading image paths in chest\_xray/{trgt\_dir} directory")

folder = "."+os.path.sep+'chest\_xray'+os.path.sep+trgt\_dir

print(f'Available classes {os.listdir(folder)}')

for root, dirs, files in os.walk(folder):

for file in files:

if file.endswith(".jpeg"):

f\_path = os.path.join(root, file)

label = f\_path.split('/')[-2]

cls\_labels.append(label)

paths.append(f\_path)

print("DONE")

return paths,cls\_labels

# Location of zipfile containing dataset

PATH = "/content/drive/MyDrive/Colab Notebooks/Kaggle\_Pneumonia\_Detection(CNN)/dataset/"

zip\_file = "archive.zip"

trainImg\_paths, y\_train = path\_extractor(PATH,zip\_file,trgt\_dir="train")

testImg\_paths, y\_test = path\_extractor(PATH,zip\_file,trgt\_dir="test")

import cv2

IMAGE\_SIZE = (150,150)

def images\_extractor(paths):

'''

Reading images from provided list and returns pre-processed images.

'''

images = []

for path in paths:

currImg\_BGR = cv2.imread(path)

b,g,r = cv2.split(currImg\_BGR)

currImg\_RGB = cv2.merge([r,g,b])

currImg = cv2.resize(currImg\_RGB,IMAGE\_SIZE)

images.append(currImg)

return images

X\_train = images\_extractor(trainImg\_paths)

X\_train = np.array(X\_train)

y\_train = np.array(y\_train)

print("Shape of train set",X\_train.shape)

print("Number of train samples",len(X\_train))

X\_test = images\_extractor(testImg\_paths)

X\_test = np.array(X\_test)

y\_test = np.array(y\_test)

print("Shape of test set",X\_test.shape)

print("Number of test samples",len(X\_test))

# Normalize the data

X\_train = X\_train / 255.0

X\_val = X\_val / 255.0

X\_test = X\_test / 255.0

from keras.preprocessing.image import ImageDataGenerator

datagen = ImageDataGenerator(

featurewise\_center=False, # set input mean to 0 over the dataset

samplewise\_center=False, # set each sample mean to 0

featurewise\_std\_normalization=False, # divide inputs by std of the dataset

samplewise\_std\_normalization=False, # divide each input by its std

zca\_whitening=False, # apply ZCA whitening

rotation\_range = 30, # randomly rotate images in the range (degrees, 0 to 180)

zoom\_range = 0.2, # Randomly zoom image

width\_shift\_range=0.1, # randomly shift images horizontally (fraction of total width)

height\_shift\_range=0.1, # randomly shift images vertically (fraction of total height)

horizontal\_flip = True, # randomly flip images

vertical\_flip=False) # randomly flip images

datagen.fit(X\_train)

# Loading Deep Learning Libraries

from keras.models import Sequential, load\_model, Model

from keras.layers import Input, Dense, Dropout, Flatten, Activation, MaxPool2D

from keras.layers import Conv2D, MaxPooling2D, BatchNormalization, GlobalAveragePooling2D, SeparableConv2D

from keras.optimizers import Adam,SGD,Adagrad,Adadelta,RMSprop

from keras.callbacks import ReduceLROnPlateau, LearningRateScheduler,ModelCheckpoint,EarlyStopping

from keras.utils import to\_categorical

# Defining parameters

input\_shape = (150,150,3)

learning\_rate = 0.00146

batch\_size = 64

epochs = 10

model = Sequential()

model.add(Conv2D(32 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu' , input\_shape = (150,150,3)))

model.add(BatchNormalization())

model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))

model.add(Conv2D(64 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))

model.add(Dropout(0.1))

model.add(BatchNormalization())

model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))

model.add(Conv2D(64 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))

model.add(BatchNormalization())

model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))

model.add(Conv2D(128 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))

model.add(Dropout(0.2))

model.add(BatchNormalization())

model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))

model.add(Conv2D(256 , (3,3) , strides = 1 , padding = 'same' , activation = 'relu'))

model.add(Dropout(0.2))

model.add(BatchNormalization())

model.add(MaxPool2D((2,2) , strides = 2 , padding = 'same'))

model.add(Flatten())

model.add(Dense(units = 128 , activation = 'relu'))

model.add(Dropout(0.2))

model.add(Dense(units = 1 , activation = 'sigmoid'))

model.compile(optimizer = "rmsprop" , loss = 'binary\_crossentropy' , metrics = ['accuracy'])

# Displaying model architecture

model.summary()

# Initializing callbacks

path = f"/content/drive/MyDrive/Colab Notebooks/Kaggle\_Pneumonia\_Detection(CNN)/model.h5"

# Saves the model in-between epochs when there is an improvement in "val\_loss"

checkpoint = ModelCheckpoint(path,

monitor="val\_loss",

mode="min",

save\_best\_only = True,

verbose=1)

# Stops training the model when no improvement in "val\_loss" is observed after set "patience"

earlystop = EarlyStopping(monitor = 'val\_loss',

min\_delta = 0,

patience = 4,

verbose = 1,

restore\_best\_weights = True)

# Monitors "val\_loss" for a set 'patience', then the learning rate is reduced by a factor specified manually.

reduce\_lr = ReduceLROnPlateau(monitor='val\_accuracy',

patience = 2, verbose=1,

factor=0.3, min\_lr=0.000001)

callbacks = [checkpoint, earlystop, reduce\_lr]

# Training the model for 10 epochs with a batch size of 32(i.e. len(X\_train)/batch\_size)

history = model.fit(datagen.flow(X\_train,y\_train, batch\_size = 32),

epochs = 10, validation\_data = datagen.flow(X\_val, y\_val),

callbacks = callbacks)

def plot\_validation\_curves(result):

result = pd.DataFrame(result)

fig, axs = plt.subplots(1,2)

result[['loss','val\_loss']].plot(figsize=(10, 3),ax=axs[0])

axs[0].set\_title('Train vs validation Loss')

axs[0].set\_xlabel('Epochs')

axs[0].set\_ylabel('Loss')

result[['accuracy','val\_accuracy']].plot(figsize=(10, 3),ax=axs[1])

axs[1].set\_title('Train vs validation Accuracy')

axs[1].set\_xlabel('Epochs')

axs[1].set\_ylabel('Accuracy')

# Performing predictions on test data

predictions = model.predict\_classes(X\_test)

predictions = predictions.reshape(1,-1)[0]

predictions[:15]

from sklearn.metrics import classification\_report,confusion\_matrix

def plot\_summary(predictions):

'''

Plots confustion matrix and classification report.

'''

# Selecting class based on higher probabilities

y\_pred\_classes = predictions

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* CLASSIFICATION REPORT \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print(classification\_report(y\_test,y\_pred\_classes),"\n\n")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* CONFUSION MATRIX \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

plt.figure(figsize=(8,8))

sns.heatmap(confusion\_matrix(y\_test,y\_pred\_classes),linewidths=.5,cmap="YlGnBu",annot=True,cbar=False,fmt='d')

plt.show()

plot\_summary(predictions)

from keras.utils import plot\_model

# Save our model diagrams to this path

model\_diagrams\_path = '/content/drive/MyDrive/Colab Notebooks/Kaggle\_Pneumonia\_Detection(CNN)/'

# Generate the plot

plot\_model(model, to\_file = model\_diagrams\_path + 'model\_plot.png',

show\_shapes = True,

show\_layer\_names = True)

# Show the plot here

img = plt.imread(model\_diagrams\_path + 'model\_plot.png')

plt.figure(figsize=(30,15))

imgplot = plt.imshow(img)

**Flask Api Code**

from flask import Flask, flash, request, redirect, url\_for, render\_template

from werkzeug.utils import secure\_filename

from tensorflow.keras.models import load\_model

import numpy as np

import os

import cv2

# Creating a Flask Instance

app = Flask(\_\_name\_\_)

IMAGE\_SIZE = (150, 150)

UPLOAD\_FOLDER = 'static\\uploads'

ALLOWED\_EXTENSIONS = set(['png', 'jpg', 'jpeg'])

app.config.from\_object(\_\_name\_\_)

app.config['SECRET\_KEY'] = '7d441f27d441f27567d441f2b6176a'

app.config['UPLOAD\_FOLDER'] = UPLOAD\_FOLDER

print("Loading Pre-trained Model ...")

model = load\_model('model.h5')

def image\_preprocessor(path):

'''

Function to pre-process the image before feeding to model.

'''

print('Processing Image ...')

currImg\_BGR = cv2.imread(path)

b, g, r = cv2.split(currImg\_BGR)

currImg\_RGB = cv2.merge([r, g, b])

currImg = cv2.resize(currImg\_RGB, IMAGE\_SIZE)

currImg = currImg/255.0

currImg = np.reshape(currImg, (1, 150, 150, 3))

return currImg

def model\_pred(image):

'''

Perfroms predictions based on input image

'''

print("Image\_shape", image.shape)

print("Image\_dimension", image.ndim)

# Returns Probability:

# prediction = model.predict(image)[0]

# Returns class:

prediction = np.argmax(model.predict(image))[0]

''' if prediction == 1:

return "Pneumonia"

else:

return "Normal"'''

return (prediction)

def allowed\_file(filename):

return '.' in filename and \

filename.rsplit('.', 1)[1].lower() in ALLOWED\_EXTENSIONS

@app.route('/', methods=['GET', 'POST'])

def home():

return render\_template('index.html')

@app.route('/upload', methods=['GET', 'POST'])

def upload\_file():

# Checks if post request was submitted

if request.method == 'POST':

'''

- request.url - http://127.0.0.1:5000/

- request.files - Dictionaary of HTML elem "name" attribute and corrospondiong file details eg.

"imageFile" : <FileStorage: 'Profile\_Pic.jpg' ('image/jpeg')>

'''

# check if the post request has the file part

if 'imageFile' not in request.files:

flash('No file part')

return redirect(request.url)

# check if filename is an empty string

file = request.files['imageFile']

if file.filename == '':

flash('No selected file')

return redirect(request.url)

# if file is uploaded

if file and allowed\_file(file.filename):

filename = secure\_filename(file.filename)

imgPath = os.path.join(app.config['UPLOAD\_FOLDER'], filename)

file.save(imgPath)

print(f"Image saved at {imgPath}")

# Preprocessing Image

image = image\_preprocessor(imgPath)

# Perfroming Prediction

pred = model\_pred(image)

return render\_template('upload.html', name=filename, result=pred)

return redirect(url\_for('home'))

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)