

C O V E N T R Y
U N I V E R S I T Y

Faculty of Engineering, Environment and Computing
School of Computing, Mathematics and Data Science

MSc. Data Science

7150CEM Data Science Project

**Enhancing Pneumonia Detection: Using Ensemble Learning in Deep
Learning-Based Models.**

Author: Aksa Annu George

SID: 13308049

Supervisor: Dr Anup Pandey

Submitted in partial fulfilment of the requirements for the Degree of Master of Science in Data Science

Academic Year: 2023/2024

Declaration of Originality

I declare that this project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet etc.) has been acknowledged by citation within the main report to an item in the References or Bibliography lists. I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Statement of copyright

I acknowledge that the copyright of this project report, and any product developed as part of the project, belong to Coventry University. Support, including funding, is available to commercialise products and services developed by staff and students. Any revenue that is generated is split with the inventor/s of the product or service. For further information please see www.coventry.ac.uk/ipr or contact ipr@coventry.ac.uk.

Statement of ethical engagement

I declare that a proposal for this project has been submitted to the Coventry University ethics monitoring website (<https://ethics.coventry.ac.uk/>) and that the application number is listed below

Signed: Aksa Annu George

Date: 8/12/2023

Please complete all fields.

First Name:	Aksa
Last Name:	George
Student ID number	13308049
Ethics Application Number	P164395
1 st Supervisor Name	Dr Anup Pandey
2 nd Supervisor Name	Dr Ashima Chopra

Abstract

Pneumonia is a deadly medical condition that affects both children and old people. The effects of pneumonia on older people and infants are devastating and it may also be life-threatening in some cases. The treatment of pneumonia and helping people recover from it is much easier if it is properly detected in its early stages. Pneumonia is mainly analysed based on X-ray scans of the chest but the analysis of the X-rays has to be accurate and the X-rays must also be clear to help in identifying pneumonia, both of these are difficult in the manual detection of pneumonia, so it has to be detected in a computer-aided manner for proper identification. In the study performed here, pneumonia is detected using deep learning methods. Three powerful pre-trained models, namely DenseNet-121, ResNet-50, and Inception-V3 are used in this study and the models are trained using the chest X-ray images in the dataset obtained from a free online source. The three deep learning models are trained and their performances in detecting pneumonia from X-ray images are tested based on their performance metrics. The best-performing deep learning model among the three models considered here was used for creating a desktop application that detects pneumonia from X-ray images. It was found that the best performance in detecting pneumonia was shown by the DenseNet-121.

Table of Contents

Abstract	3
Acknowledgments.....	6
1 Introduction	7
1.1 Aim.....	8
1.2 Research questions	9
1.3 Objectives.....	9
1.4 Report structure	9
2 Literature review	10
2.1 Data Augmentation.....	10
2.2 Image Classification	10
2.3 ResNet50	11
2.4 Inception V3	12
2.5 DenseNet121	14
2.6 Related work	15
2.6.1 Machine learning methods for the detection of pneumonia.....	15
2.6.2 Deep learning methods for the detection of pneumonia	17
2.7 Research Gap	21
3 Methodology	22
3.1 Tools Used.....	22
3.2 Dataset	22
3.3 Data Preprocessing	22
3.4 Training and validation split	23
3.5 Deep learning models.....	24
3.6 Performance Evaluation	28
4 Results	31
4.1 Accuracy.....	31
4.2 Precision.....	31
4.3 Recall	31
4.4 F1 Score	32
4.5 Confusion Matrix	32
4.6 Graphical user interface (GUI)	35
5 Project Management	37

5.1 Project Schedule.....	37
5.2 Risk Management	37
5.3 Quality Management.....	38
5.4 Social, Legal, Ethical and Professional Considerations.....	38
6 Discussion and evaluation.....	39
6.1 Discussion	39
6.2 Critical evaluation	40
7 Conclusion and future enhancements	41
7.1 Conclusion	41
7.2 Future enhancements.....	41
8 Student Reflections	42
References	43
Appendices.....	47
Appendix A	47
Appendix B	48
Appendix C	49
Appendix D.....	66

Acknowledgments

I would like to show my gratitude towards my supervisor Dr Anup Pandey whose advice helped me in completing the project successfully.

1 Introduction

Pneumonia is a significant public health concern, impacting individuals across various age groups and presenting substantial risks to respiratory health. This common and potentially life-threatening illness affects the air sacs in the lungs, leading to the accumulation of fluid that impedes normal breathing. It can be caused by a range of microorganisms, including bacteria, fungi, and viruses, with heightened risks observed among young children, infants, and older individuals. The global burden of pneumonia is alarming, especially in vulnerable populations. Statistics from Gabruseva, Poplavskiy, and Kalinin (2020) reveal the devastating toll of this disease, accounting for approximately 16% of deaths among children under the age of 5. Additionally, the World Health Organization (2021) reports that pneumonia affects over 1,400 children per 100,000 annually, equating to approximately 1 in 71 children being impacted each year. In the United States, the incidence of pneumonia is staggering, with over one million hospital admissions attributed to pneumonia annually, resulting in nearly 500,000 fatalities (Rajpurkar et al., 2017). The emergence of the COVID-19 pandemic has added a new dimension to the impact of pneumonia. Individuals suffering from pneumonia concurrently infected with the COVID-19 virus face heightened challenges and increased health risks. The dual burden of these respiratory illnesses can severely compromise an individual's health and may lead to more severe health outcomes. Given the potential for pneumonia to significantly impact the lungs over an extended period, the co-occurrence of pneumonia and COVID-19 poses a particularly grave threat to respiratory health and overall well-being. The repercussions of pneumonia, especially among children, are far-reaching, potentially leading to long-term impairment of lung function and persistent respiratory issues. The adverse effects on lung health can be enduring, resulting in chronic breathing difficulties and ongoing challenges with lung capacity. As such, the identification and timely treatment of pneumonia are critical in preventing substantial and lasting damage to an individual's health. Early intervention can help mitigate the potential long-term consequences of pneumonia and minimize its impact on respiratory function. Effective management of pneumonia necessitates a multi-faceted approach. Early diagnosis is paramount to promptly initiate appropriate medical interventions and prevent the progression of the illness. Treatment strategies typically involve the use of antibiotics, antiviral medications, and supportive care to address symptoms and alleviate respiratory distress. In severe cases, hospitalization may be required to provide intensive care and respiratory support to affected individuals. Preventive measures are also vital in mitigating the impact of pneumonia. Vaccination, particularly for diseases such as influenza and pneumococcal infection, plays a crucial role in reducing the incidence of pneumonia and its associated complications. Health promotion efforts aimed at fostering good respiratory hygiene, maintaining a healthy lifestyle, and avoiding exposure to known respiratory pathogens are fundamental in preventing the onset of pneumonia. Furthermore, ongoing research and advancements in medical science are pivotal in improving the understanding, prevention, and treatment of pneumonia. Continued investment in research, development of innovative treatment modalities, and heightened awareness about the risks and consequences of pneumonia are essential in combatting this pervasive and potentially life-threatening respiratory illness.

One major way in which pneumonia can be determined is by using X-rays. X-rays are generated using a of high-energy electromagnetic radiation and when this radiation is focused on a particular

part of the human body, an image of the internal condition of the body part can be seen as an image. The image is formed based on the passage of radiation through a body part. An X-ray of the chest can be generated by focusing the radiations in the chest and this can be used for determining if a person suffers from pneumonia or not. The region of the X-rays in which the opacity is high shows the presence of pneumonia (Franquet, 2018). Clinical signs and medical history of a person can be combined with the nature of the X-rays to conclusively determine if a person is affected by pneumonia or not. But still, it cannot be said that the detection of pneumonia from an X-ray is completely accurate as different factors like bleeding, overload of fluid in lungs, lung cancer, loss of volume of the lungs, or changes to the lungs due to surgery or being exposed to radiation. Even if a number of X-rays can be generated in different positions (figure (1)), the identification of pneumonia may still be difficult as several variables, including the patient's position and the depth of their breath, can modify the appearance of a chest X-ray further complicating its interpretation (Kelly, 2012).

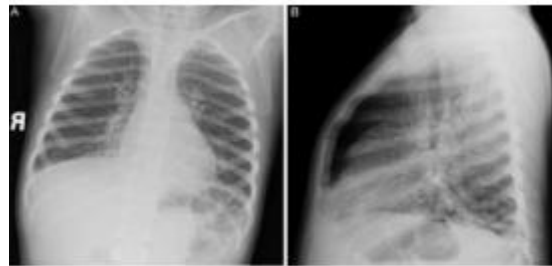


Figure (1): chest X-rays in different positions (Khan, Zaki and Ali, 2021)

Several other problems are also associated with the inability to identify pneumonia from X-ray images like the fact that as the analysis of the X-ray images is done using the human eye, errors are bound to occur and also the analysis of one doctor may be different from the analysis of another doctor (Neuman et al., 2011). So for the accurate detection of pneumonia from X-ray images, an automatic method is required with the help of computers.

In recent years several problems associated with the automatic analysis or the prediction based on some kind of data are carried out using machine learning and deep learning algorithms. The methods based on machine learning and deep learning algorithms have been used for the identification and classification problems based on medical images (Rakhlin et al., 2018). So machine learning and deep learning methods can be used for the detection of pneumonia from X-ray images. In the study performed here deep learning methods are used for the detection of pneumonia from X-ray images.

1.1 Aim

The main aim of the study is to create a system that is able to detect if a person has pneumonia or if they are normal based on X-ray images.

1.2 Research questions

Three pre-trained deep learning models, DenseNet-121, ResNet-50, and Inception-V3, are used in this study. So for the study performed here, the main research questions are:

- Which deep learning model among the DenseNet-121, ResNet-50, and Inception-V3 will have the better performance in detecting pneumonia from X-ray images?
- Will the performance of the best-performing model in this study in detecting pneumonia be greater than the performance of an existing deep learning-based pneumonia detection system?

1.3 Objectives

The main objectives of the study are given below:

- Transform the dataset images into a format compatible with training the deep learning models.
- Determine the performance metrics of the three deep learning models used here, separately.
- Determine the deep learning model which has the best performance in pneumonia detection.
- Create a desktop application that identifies pneumonia based on an X-ray image given as input.
- Compare the performance of the best-performing deep learning model used here to existing deep learning-based pneumonia detection systems.

1.4 Report structure

Section 1 of this report consists of the reasons for developing a deep learning-based pneumonia detection system. This section also discusses the main objectives and aims of the study. Section 2 discusses the review of the main literature associated with existing pneumonia detection systems. Section 3 discusses the main methods used in the development of the deep learning-based pneumonia detection system. Section 4 discusses the results of the study performed here. Section 5 discusses the main objectives achieved and the critical evaluation of the study. Section 6 discusses the details about the management of the project. Section 7 discusses the concluding remarks about the main methods used in the study and the results. Section 8 contains the reflection of the student on implementing a deep learning-based pneumonia detection system.

2 Literature review

2.1 Data Augmentation

Initial evidence of the utility of data augmentation dates back to basic transformations like horizontal flipping, adjustments in colour space, and random cropping. These transformations encapsulate various invariances previously highlighted, which pose challenges in image recognition tasks. The survey encompasses a range of augmentations, including geometric transformations, colour space alterations, kernel filters, image blending, random erasing, augmentations in feature space, adversarial training, GAN-based augmentation, neural style transfer, and meta-learning strategies (Shorten and Khoshgoftaar, 2019).

Data augmentation is a pivotal technique in enhancing the robustness of datasets, encompassing two primary categories: data warping and oversampling methods. This approach holds immense promise for the future, especially with the emergence of search algorithms that effectively integrate data warping and oversampling techniques, thereby offering significant potential for improving dataset robustness. Furthermore, the layered architecture of deep neural networks presents numerous opportunities for leveraging data augmentation to enhance the performance and reliability of machine learning models. While the majority of existing augmentations predominantly operate at the input layer, there is growing exploration into augmentations derived from hidden layer representations within neural networks. Notably, certain methods, such as DisturbLabel, extend their influence to the output layer, showcasing the potential for augmentations to exert an impact across multiple layers of a deep neural network. This exploration of the space encompassing intermediate representations and the label space within neural networks remains relatively unexplored, presenting captivating possibilities for advancing the capabilities of data augmentation techniques. Although this review primarily focuses on applications related to image data, it is essential to recognize that the principles and methodologies discussed within this context hold the potential to be extrapolated to other domains of data, including text, audio, and time-series data. By leveraging the foundational concepts of data augmentation and adapting them to diverse data domains, it becomes feasible to improve the robustness and generalization capabilities of machine learning models across a broad spectrum of applications (Shorten and Khoshgoftaar, 2019).

2.2 Image Classification

In the realm of computer vision, image classification serves as a critical endeavour that seeks to emulate human perception by instructing computers through data. It involves categorizing images into predefined groups according to their unique visual attributes. Historically, image classification techniques have been integral to the larger domain of artificial intelligence, specifically within the scope of machine learning. Machine learning is composed of two principal components: a feature extraction module, which is tasked with identifying essential image characteristics such as edges and textures, and a classification module, which allocates categories based on the features that have been extracted. (Manoj Krishna et al., 2018).

2.3 ResNet50

RESNET, which stands for residual neural network, represents a significant advancement in addressing the challenge of improving network feature extraction by introducing the concept of cross-layer feature fusion. This innovative approach is designed to enhance the performance of deep neural networks as their depth increases, thereby overcoming the limitations associated with network degradation. The introduction of RESNET marks a crucial milestone in the evolution of deep learning models, offering valuable insights into the potential of cross-layer feature fusion to elevate the capabilities of neural networks. The commendable performance of RESNET has been substantiated through comprehensive testing, wherein the research team conducted experiments using increasingly deep RESNET models and compared their performance with that of various other deep learning models. The results of these tests unequivocally showcased the superior classification performance of RESNET in comparison to other models. Furthermore, the findings underscored the positive correlation between the depth of the network and its ability to achieve heightened accuracy, providing compelling evidence of the efficacy of RESNET as a powerful tool for advancing the field of deep learning. It is noteworthy to acknowledge that RESNET, also known as a residual network, was conceptualized and introduced by the Kaiming team at the prestigious Microsoft Research Institute, and is recognized by the same name in its original Chinese nomenclature. This groundbreaking innovation from the Kaiming team represents a pivotal contribution to the field of deep learning, distinctly aiming to overcome the persistent challenge of network degradation that manifests in the form of elevated training error rates in deeper neural networks (Liang, 2020). The underlying motivation driving the design of RESNET revolves around surmounting the inherent limitations encountered in training deeper neural networks, ultimately seeking to mitigate the phenomenon of network degradation. By introducing the concept of residual connections, which facilitate the propagation of information across network layers while mitigating the impact of vanishing gradients, RESNET offers an elegant and effective solution to enhance the performance and scalability of deep neural networks.

In a particular scenario, the utilization of the ResNet50 model transcended traditional machine learning approaches and ventured into the domain of deep learning, epitomizing its flexibility and adaptability across diverse applications. Initially, the ResNet50 model assumed the role of a feature extractor, denoted as a pre-trained ResNet50 network (Figure 2), signifying its utilization in a manner akin to traditional machine learning methodologies. Subsequently, in a distinct scenario, the ResNet50 model underwent fine-tuning, assuming the mantle of a deep-learning model specifically engineered for image classification tasks, underscoring its remarkable versatility across disparate use cases. Acknowledging the intrinsic computational demands associated with both feature extraction and learning processes, particularly when dealing with large images, a strategic decision was made to further reduce the size of the selected image patches through a downsampling process, aptly compressing them by a factor of four. This meticulous approach served to optimize computational resources, while preserving the integrity of the visual data pertinent to the ResNet50 model's subsequent analytical processes (Ali et al., 2019). In both instances of the ResNet50 network implementation, the network was furnished with the downscaled image patches, furnishing it with the requisite inputs for subsequent feature extraction and image classification endeavors. Notably, novel architectural innovations were introduced, harnessing the ResNet50 model's convolutional layers to adeptly extract feature values for each image, thus laying the foundational groundwork for robust image analysis and interpretation. These innovative architectural adaptations

extended beyond the scope of the original ResNet-50 model, necessitating the replacement of its standard 1000-node fully connected softmax layer, traditionally tasked with image categorization (Hamlili et al., 2022). Steering the ResNet-50 model towards unprecedented territory, these novel architectural designs culminated in an evolution of its image classification framework, wherein the classification phase featured the integration of three fully connected layers, supplemented by three dropout layers and a decisive softmax layer, symbolizing a departure from the conventional image classification paradigms. The first two fully connected layers within this innovative design assumed dimensions of 200 and 300, respectively, with the size of the final layer dynamically contingent upon the specific number of categories present within the dataset under analysis. Notably, the strategic integration of dropout layers, each characterized by distinct dropout ratios of 0.12, 0.2, and 0.15, was implemented to effectively mitigate overfitting and augment the overall performance of the ResNet-50 model within the context of image classification tasks (Hamlili et al., 2022). Underpinning the image classification architecture's efficacy was the strategic adoption of the softmax activation function, serving as a pivotal component in the network's decision-making process, effectively delineating and categorizing images with a high degree of accuracy and precision. Through the seamless integration of these diverse architectural elements, the ResNet-50 model transcended its conventional boundaries, emerging as a potent vehicle for image analysis and classification, fortified by a nuanced and adaptive architecture tailored to the unique demands of image processing and interpretation.

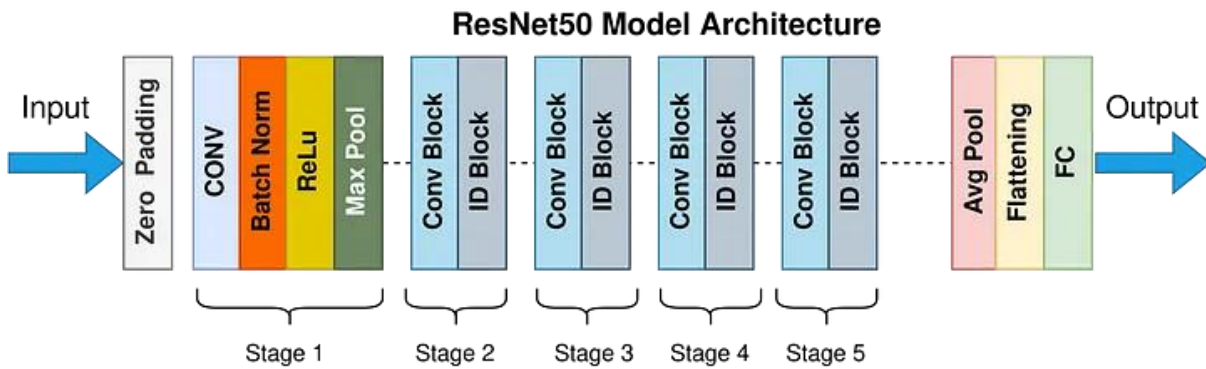


Figure (2):Resnet-50 Architecture (Mukherjee, 2022)

2.4 Inception V3

For Inception-v3, the architectural configuration consists of a sequential arrangement of 3 Inception A modules, 5 Inception B modules, and 2 Inception C modules. The standard input image size for Inception-v3 is 299x299 pixels, but in your specific dataset, the images were of a different size, measuring 224x224 pixels. Notably, you made the deliberate choice not to resize the images to 299x299 during both the training and testing phases. This decision solely affected the dimensions of the feature maps generated while leaving the number of channels unaltered, and remarkably, the results obtained remained satisfactory (Guan et al., 2019).

The development of Inception-v3 represents a significant endeavour aimed at addressing the critical challenges of computational efficiency and parameter limitations encountered in real-world applications, particularly in the realm of deep learning and image classification. Notably, Inception-v3 stands out for its remarkable performance, despite accommodating a larger input image size of 299x299 pixels, which represents a substantial 78% increase compared to the 244x244 pixel input

size of VGGNet (Cao et al., 2021). This enhancement in input resolution is accompanied by notable gains in processing speed, positioning Inception-v3 as a compelling solution that outpaces VGGNet in terms of computational efficiency while delivering exceptional performance. The ubiquity of convolutional neural networks (CNNs) in the domain of image classification underscores their growing prominence as a foundational technology for addressing complex visual recognition tasks. Inception-v3, as a prominent CNN-based model, initially designed to proficiently classify an extensive range of 1000 classes amidst millions of images, uniquely exemplifies adaptability by offering the versatility to be easily retrained for custom image classification tasks. This adaptability is a key strength of Inception-v3, underpinning its versatility and utility in diverse applications, including the classification of apparel items and a multitude of other visual recognition challenges. It is this inherent flexibility that cements Inception-v3's position as a formidable asset in image classification endeavors, empowering researchers and practitioners to effectively leverage its high classification accuracy across a wide array of practical scenarios. The work of Tio (2019) brings to the forefront the potential for leveraging CNNs, such as Inception-v3, in the domain of face shape classification. By harnessing the inherent capabilities of CNNs, this research presents a tantalizing prospect of achieving enhanced accuracy in face shape classification, all without the intricacies and complexities associated with manual feature selection and extraction. By simplifying the model training process, CNNs, including Inception-v3, offer a streamlined and efficient approach to face shape classification, underpinned by the robust capabilities of deep learning and the innate adaptability of Inception-v3.

For Inception-v3, the architectural configuration consists of a sequential arrangement of 3 Inception A modules, 5 Inception B modules, and 2 Inception C modules (Figure(3)). The standard input image size for Inception-v3 is 299x299 pixels, but in your specific dataset, the images were of a different size, measuring 224x224 pixels. Notably, you made the deliberate choice not to resize the images to 299x299 during both the training and testing phases. This decision solely affected the dimensions of the feature maps generated while leaving the number of channels unaltered, and remarkably, the results obtained remained satisfactory (Guan et al., 2019). The Inception modules are intelligently crafted convolutional building blocks with a dual objective: extracting unique features from input data while efficiently managing the number of model parameters. These modules involve the integration of multiple convolutional and pooling layers that operate in parallel. To maintain a compact parameter count, they incorporate smaller convolutional layers with dimensions such as 3x3, 1x3, 3x1, and 1x1. (Guan et al., 2019).

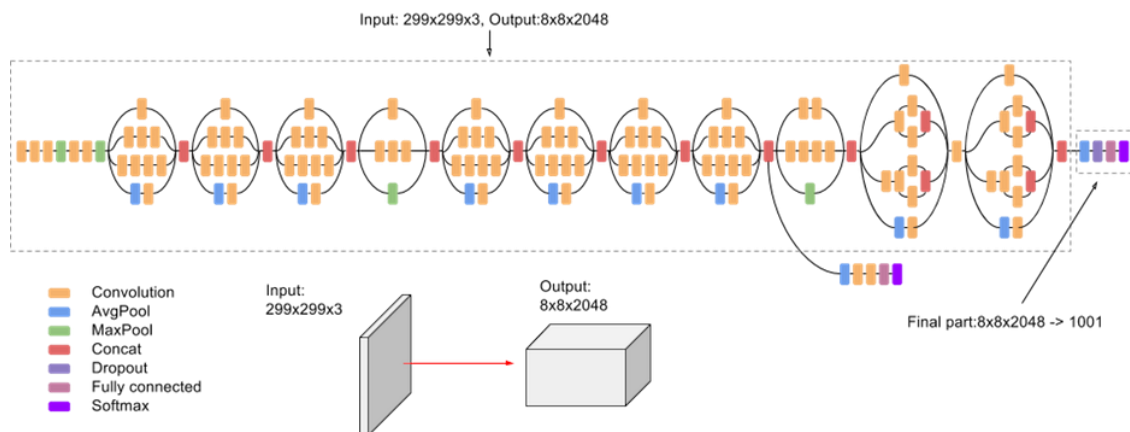


Figure (3): Inception-V3 Architecture (OpenGenus IQ: Computing Expertise & Legacy, 2021)

2.5 DenseNet121

DenseNet, short for Dense Convolutional Network, is an architectural design focused on enhancing the depth of deep learning networks while optimizing their training efficiency. It achieves this by establishing dense connections between network layers. In DenseNet121, each layer is connected to all non-sequential layers that are deeper within the network. In other words, the first layer is connected to the second, third, fourth, and so on, while the second layer is linked to the third, fourth, fifth, and so forth (Figure(4)). This connectivity scheme is implemented to facilitate maximum information flow between the network's layers. To maintain a feed-forward nature, every layer receives input from all preceding layers and provides its feature maps to all subsequent layers in the network. This architecture is designed to promote deep network learning while effectively utilizing feature information at each layer (Vellaichamy et al., 2021).

DenseNet121 has emerged as a standout performer, showcasing exceptional capabilities while operating with reduced memory consumption and processing power, presenting a compelling proposition when compared to other state-of-the-art techniques. This system holds immense promise across a diverse spectrum of applications, ranging from traffic monitoring and intelligent transportation systems to augmenting driving functionalities in vehicles. The inherent strength of DenseNet-121 lies in its ability to effectively address the challenge of vanishing gradients, yielding a multitude of beneficial outcomes that significantly elevate its prowess in the domain of deep learning. Foremost among the advantages offered by DenseNet-121 is its adeptness in mitigating the training burden often associated with deep learning models. By alleviating the impact of vanishing gradients, this model streamlines the training process, enhancing both the efficiency and efficacy of deep learning endeavours. Furthermore, its capacity to facilitate feature reuse contributes to the optimization of overall efficiency, offering a profound impact on the model's performance. Additionally, DenseNet-121 stands out for its judicious utilization of parameters, effectively reducing parameter usage in comparison to other prevailing deep learning models, positioning it as a resource-efficient option within the landscape of deep learning methodologies. Amidst the numerous merits of DenseNet-121, its implementation posed a notable challenge, specifically stemming from the absence of a well-labelled training dataset. Overcoming this hurdle necessitated innovative approaches and robust methodologies to realize the model's full potential, underscoring the significance of developing effective strategies for dataset creation and curation to maximize the efficacy of DenseNet-121 in practical applications. Delving into the architectural intricacies of DenseNet-121, its structure comprises four dense blocks, each interspersed with a transition layer. These transition layers play a pivotal role in implementing down-sampling on the feature maps, achieving this through a combination of a 1x1 convolution layer and a 2x2 average pooling layer. Within the dense blocks, multiple convolutional layers are strategically arranged in a sequential manner, establishing interconnectedness between distant layers. This intricate arrangement serves to intricately weave together the network's layers, fostering a high degree of interconnectivity and information flow, thereby contributing to the model's exceptional performance. Furthermore, to bolster the network's non-linearity, DenseNet-121 leverages the Rectified Linear Unit (ReLU) activation function, enhancing its capacity to capture complex, non-linear relationships within the data. Ultimately, the architecture culminates with a fully connected layer that incorporates a softmax function, enabling the model to make predictions regarding the probability of an image belonging to a specific class, underpinning its prowess in image classification tasks (Albelwi, 2022).

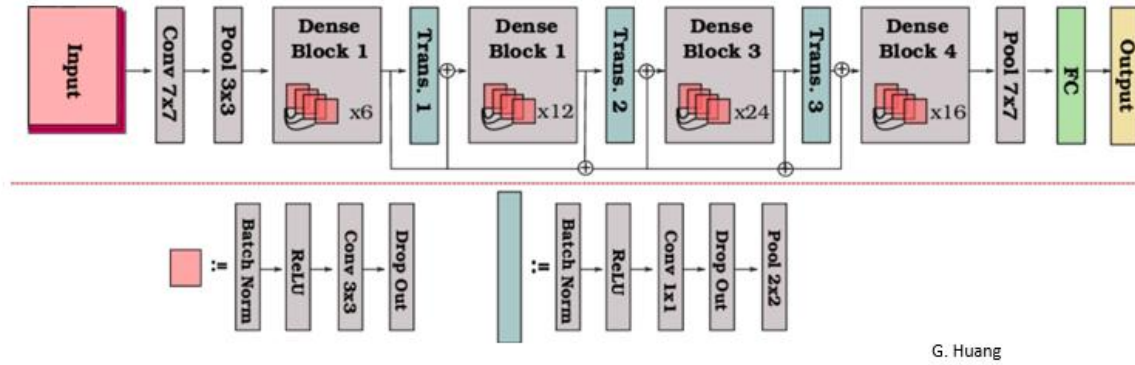


Figure (4): Densenet-121 Architecture (pluralsight, 2023)

2.6 Related work

The literature review associated with the existing systems that pneumonia using deep learning models were studied and the review associated with each of those methods is discussed in this part.

2.6.1 Machine learning methods for the detection of pneumonia

A Convolutional Neural Networks (CNNs) based feature extraction was used for the detection of pneumonia in chest X-ray images. The study by (Varshni et al., 2019) focuses on the use of pre-trained CNN models as feature extractors, followed by different classification techniques to distinguish between abnormal (pneumonia) and normal chest X-rays. The dataset employed for this research is ChestX-ray14, containing a substantial number of frontal chest X-ray images labeled with various thoracic diseases. Natural Language Processing (NLP) was used to extract these labels from associated radiological reports. The study proposes a pneumonia detection system using the DenseNet-169 architecture, which is divided into three stages: preprocessing, feature extraction, and classification. The combination of CNN-based feature extraction and supervised classifiers, particularly the support vector machine (SVM), was found to be highly effective in classifying the X-ray images. The research explores various pre-trained CNN models such as VGGNets, Xception, ResNet-50, and DenseNets, in combination with different classifiers. The results show that ResNet-50 and DenseNets, particularly DenseNet-169, outperformed other models when used with default SVM parameters. The study also involves hyperparameter tuning, with DenseNet-169 emerging as the best feature extractor when combined with optimal SVM parameters. While the results were promising, the study acknowledges certain limitations, including the lack of patient history, the use of only frontal chest X-rays, and the computational intensity of the model. Despite these limitations, the proposed model demonstrates better performance in pneumonia detection compared to previous work. This study provides valuable insights into the effectiveness of CNN-based feature extraction and classification for pneumonia detection in chest X-ray images. The use of pre-trained models and SVM classifiers, as well as the selection of DenseNet-169, contributes to the model's success in this critical medical imaging task.

A deep learning model based on the VGG16 architecture by (Sharma and Guleria, 2023) for the detection and classification of pneumonia in chest X-ray (CXR) images using two distinct datasets. The primary dataset, obtained from Kaggle, consists of 5,856 CXR images of pediatric patients aged 1 to 5 years, categorized into pneumonia and normal lung images. The results of this model reveal an impressive accuracy of 92.15%, along with a precision of 0.9428, recall of 0.9308, and F1-score of 0.937. The secondary dataset, consisting of 6,436 images of pneumonia, normal lungs, and COVID-19 cases, is also utilized, where the model exhibits a remarkable accuracy of 95.4%, precision of 0.954, recall of 0.954, and an F1-score of 0.954. The research findings indicate that the proposed model with the VGG16 architecture outperforms other classification models, including SVM, KNN, Random Forest, and Naïve Bayes, in terms of accuracy and overall performance. The limitation of this study emphasizes the importance of early pneumonia detection, particularly in regions with limited healthcare infrastructure and unsanitary living conditions. However, this paper primarily focuses on the detection and classification of pneumonia in CXR images.

Usual Interstitial Pneumonia(UIP) in Transbronchial Biopsies(TBB) is detected using machine learning in the study by(Pankratz et al., 2017). The research gathered 283 TBB (Transbronchial Biopsy) samples from 84 subjects suspected of having Interstitial Lung Diseases (ILDs). RNA sequencing was conducted through exome enrichment on these samples. They were classified as either UIP (Usual Interstitial Pneumonia) or non-UIP based on expert pathology reviews of various lung biopsies from the same individuals. Using an elastic net penalty logistic regression model, the study trained on 170 TBB samples from 53 subjects, employing 169 genes as features. This model was subsequently tested on an independent set of 113 TBB samples from 31 subjects. The outcomes displayed the model's capacity to differentiate between UIP and non-UIP conditions in single TBB samples with an Area Under the Receiver Operator Characteristic Curve(AUC) of 0.86, featuring 86% specificity and 63% sensitivity. Notably, the performance elevated to an AUC of 0.92 when combining three to five TBB samples per subject at the RNA level for testing. Moreover, the study revealed that classifier accuracy was not contingent on the expression of alveolar-specific genes, suggesting that thorough alveolar sampling might not be imperative for the genomic approach. However, a limitation acknowledged in the study was the restricted and biased cohort size, which lacked adequate representation of certain ILD subtypes and patients with unclear or inconclusive pathology.

Based on a non-invasive measure by (Effah et al., 2022) investigates the application of machine learning models in the prediction of pneumonia based on a dataset comprising 535 patients with 45 features each, encompassing biomarkers, laboratory parameters, and physical attributes. The research comprises three distinct experiments: feature selection, analysis of the imbalanced original dataset, and examination of the SMOTE data, with the primary objective of evaluating the effectiveness of eight machine learning models in pneumonia prediction. Notably, biomarkers such as C-reactive protein and procalcitonin emerge as pivotal in discriminating between pneumonia cases. Ensemble machine learning models, specifically Random Forest (RF) and XGBoost, exhibit exceptional performance on the original dataset, with RF achieving an accuracy of 92.0%, precision of 91.3%, recall of 96.0%, and an F1-Score of 93.6%, while XGBoost attains an accuracy of 90.8%, precision of 92.6%, recall of 92.3%, and an F1-Score of 92.4%. Furthermore, AUC values for RF and XGBoost are 0.96 and 0.97, respectively, underlining their efficacy. When considering the SMOTE dataset, both RF and XGBoost maintain robust predictive performance, with F1-scores of 92.0% and 91.2% and AUC values of 0.97 for both models. A notable limitation of this study lies in its reliance on retrospective data, potentially introducing biases and constraints inherent to such data sources. Despite this limitation, the research highlights the potential of machine learning models, particularly RF and XGBoost, in accurately predicting pneumonia, emphasizing the importance of biomarkers and the need for further exploration in prospective clinical settings.

In the context of pneumonia diagnosis, especially amid the COVID-19 pandemic, the study by (Ortiz-Toro et al., 2022) explores the potential of three textural image characterization methods—radiomics, fractal dimension, and superpixel based histon as biomarkers for training Artificial Intelligence (AI) models to detect pneumonia in chest X-ray images. Exploring datasets from pediatric chest X-rays and a repository focused on COVID-19, the study employs K-Nearest Neighbors, Support Vector Machine, and Random Forest algorithms. The results showcase the effectiveness of the tested methods, with superpixel-based histon and fractal dimension standing out for their enhanced accuracy and F-score compared to the classical radiomics approach. The best models achieved impressive accuracy and sensitivity, 91.3% accuracy with 90.5% sensitivity for fractal dimension in the superpixels histon dataset. These findings underscore the potential of textural image characterization as a reliable tool for automated pneumonia diagnosis, presenting opportunities for improved AI-assisted triage and management. While the study exhibits promising results in leveraging textural image characterization for pneumonia detection, a major limitation lies in the potential interpretability challenges associated with these methods. The intricate nature of features extracted through radiomics, fractal dimension, and superpixel-based histon may hinder the intuitive understanding of the AI models' decision-making processes.

2.6.2 Deep learning methods for the detection of pneumonia

The research paper by (Kumar et al., 2023) presents a novel approach for pneumonia diagnosis through the evaluation of deep learning models, namely VGG16, VGG19, RESNET-50, and RESNET-101, using X-ray image datasets. The study's primary objective of the study is to identify the most effective deep learning model for the development of an application capable of distinguishing between pneumonia-affected and normal X-ray images. The investigation entails comprehensive analysis and comparison of these models, leveraging datasets of lung X-ray scans obtained from Kaggle. By conducting experiments on this dataset, the study reports the accuracy values of the deep learning models, revealing that VGG19 achieved an accuracy of 93.12%, followed closely by VGG16 with an accuracy of 92.2%. RESNET-50 and RESNET-101 achieved accuracies of 74.29% and 74.21%, respectively. Notably, the findings indicate that RESNET-50 performed exceptionally well, outperforming the other models when coupled with default parameter values of the support vector machine (SVM) classifier. These results are promising as they showcase the potential of deep learning models in the field of medical image analysis and, more specifically, in enhancing the accuracy of pneumonia detection. One significant limitation pertains to the study relying exclusively on frontal chest X-rays, overlooking the potential insights that lateral view chest X-rays might provide for a more accurate diagnosis. Lastly, the computational demands of the model are notably high due to the extensive use of convolutional layers, which may render it impractical for applications without access to substantial computational resources. Nevertheless, the study's findings have implications for advancing pneumonia diagnosis and may inspire further research in the realm of medical image analysis, particularly the exploration of patient-specific factors and the integration of multiple views for more robust diagnostics.

Machine learning and deep learning methods are used for the detection of pneumonia in the study by (Al Mamlook, Chen and Bzizi, 2020). The dataset comprises 5,856 chest X-ray images, including 1,583 normal cases and 4,273 cases diagnosed with pneumonia. To prepare the images for analysis, several preprocessing steps are implemented, involving resizing, dimension reduction, and image convolution. The study undertakes a comparative analysis of seven distinct machine learning models: DT, RF, KNN, AdaBoost, Gradient Boost, XGBoost, and Convolutional Neural Networks (CNN).

The outcomes indicate that CNN showcases the highest performance metrics, achieving an accuracy of 98.46%, F1-score of 98.95%, and an AUC of 98.48%. Following closely, RF attains 97.61% accuracy, 97.83% F1-score, and 97.72% AUC. The study draws the conclusion that CNN excels over the other models in accurately categorizing pneumonia based on chest X-ray images. Additionally, it suggests the potential utility of CNN as a diagnostic tool for pneumonia and other lung-related ailments. However, the study does acknowledge certain limitations, particularly its reliance on a singular dataset. Additionally, the research does not validate the model's performance across different datasets or assess its efficiency in analyzing more intricate medical images.

Deep learning is used for the detection of pneumonia in the study by (Racic et al., 2021). In this study, a deep learning algorithm based on CNN was employed to classify chest X-ray images, discerning the presence or absence of pneumonia. The dataset encompassed 5856 chest X-ray images, segregated into distinct subsets for training, validation, and testing. Preprocessing of the images involved converting them to grayscale, resizing to 200x200 pixels, and normalizing the pixel values. The CNN architecture comprised four convolutional layers, two max pooling layers, two dropout layers, and a single fully connected layer. The model underwent training for 200 epochs and was subsequently assessed using the test set. The findings revealed that the model achieved an accuracy of 88.90%, demonstrating similarity to outcomes observed in previous studies leveraging analogous methodologies. Analysis through the confusion matrix displayed that the model accurately classified 334 out of 381 images depicting pneumonia and 187 out of 205 images without pneumonia. However, the study acknowledged certain limitations, notably the potential for overfitting due to the dataset's size and imbalance.

Pneumonia is detected and classified using deep learning in the study by (El Asnaoui, Chawki and Idri, 2021). The study utilized a publicly available dataset featuring 5856 images extracted from chest X-ray and computed tomography (CT) scans, encompassing normal and pneumonia cases. The research involved fine-tuning the upper layers of eight distinct deep learning models, namely VGG16, VGG19, DenseNet201, Inception_ResNet_V2, Inception_V3, Resnet50, MobileNet_V2, and Xception. The performance of these models was evaluated based on various metrics including accuracy, sensitivity, specificity, precision, and F1 score. Results revealed that Resnet50, MobileNet_V2, and Inception_ResNet_V2 attained the highest accuracy, exceeding 96%, along with an F1 score of over 0.96 among the examined models. Conversely, Xception exhibited the lowest accuracy of 69.03% and F1 score of 0.79. Additionally, the study presented detailed insights such as confusion matrices and training as well as validation curves for each model. However, it's important to note that the study faced limitations owing to the relatively small and imbalanced nature of the dataset, which could potentially impact the models' generalization and robustness.

Transfer learning and CNN are used for the detection pneumonia in the study by (Puneet Gupta, 2021). The dataset comprises 5,863 x-ray images classified into normal and pneumonia cases, distributed across training, validation, and testing subsets. The study assesses the performance of three distinct CNN models: VGG16, VGG19, and a self-constructed CNN model. Results indicate that the customized CNN model achieves the highest validation accuracy at 93%, with a training accuracy of 99%. Following this, VGG16 demonstrates 92% validation accuracy and 97% training accuracy, whereas VGG19 reaches 89% validation accuracy and 95% training accuracy. However, a notable limitation of the research is the extensive training time required by the traditional CNN model.

Deep learning is used for the detection of pneumonia from chest x-ray images in the study by (Barhoom, Samy and Naser, 2022). The dataset encompassed 5,863 images featuring 14 distinct diseases, inclusive of bacterial and viral pneumonia, sourced from the ChestX-ray14 dataset. The study applied four unique convolutional neural network architectures: VGG16, VGG19, ResNet50,

and a self-designed CNN. Findings revealed that the customized CNN excelled, achieving the highest accuracy at 98.2% for pneumonia identification, followed by ResNet50 at 97.8%, VGG19 at 97.6%, and VGG16 at 97.4%. Moreover, the custom CNN displayed superior performance in sensitivity, specificity, precision, and F-score compared to the other models. The study successfully showcased the practicality and efficacy of utilizing deep learning for diagnosing pneumonia from chest x-ray images. Nonetheless, the conventional CNN, despite its superior performance, exhibited extensive training time.

An automated pneumonia diagnosis system by (Sunil L. Bangare et al., 2022) employs deep CNNs to analyze X-ray scans. Leveraging the X-ray scan dataset comprising 5863 scans, the study demonstrates the efficacy of the network model with diverse metrics, including accuracy, recall, precision, and AUC ranking. The proposed framework attains a remarkable 91% accuracy in categorization. To enhance model efficiency, various optimization techniques, such as stochastic gradient descent, Adagrad, and Adam optimizer, were explored, with the customized VGG16 model excelling, particularly when compared to Adam. Although the research indicates the successful diagnosis of pneumonia, the paper has certain limitations. It primarily focuses on binary classification (pneumonia-affected and unaffected), and future endeavors could extend the model to handle multi-class X-ray image classification. Moreover, further improvements may involve the incorporation of more advanced feature extraction techniques based on recent deep learning models designed for biomedical image segmentation, ultimately advancing the model's performance and capabilities.

A computer-aided diagnosis system for automated pneumonia detection from chest X-ray images was developed in the study by (Jain et al., 2020). By manipulating deep transfer learning, an ensemble of three convolutional neural network models - GoogLeNet, ResNet-18, and DenseNet-121 - is constructed to classify X-ray images into "Pneumonia" and "Normal" categories. A novel weighted average ensemble approach, based on scores from four evaluation metrics (precision, recall, f1-score, and AUC), replaces traditional empirical weight assignment. The proposed method undergoes rigorous testing on two publicly available pneumonia X-ray datasets, resulting in impressive accuracy rates of 98.81% and 86.85% and sensitivity rates of 98.80% and 87.02% for the Kermany and RSNA datasets, respectively. The model's domain-independence underscores its applicability in various computer vision tasks. However, one potential limitation of this study is the exclusive use of publicly available datasets, which may not fully represent the clinical diversity encountered in real-world scenarios.

The approach in the study by (Islam et al., 2019) for the automatic detection of pneumonia in X-ray images by combining Compressed Sensing (CS) with deep learning techniques. The proposed method demonstrates impressive results, achieving a high prediction accuracy of 97.34% and significantly enhancing the reconstruction quality of X-ray images, as measured by Peak-Signal-to-Noise-Ratio (PSNR) and mean Structural SIMilarity (SSIM), outperforming state-of-the-art methods. The study employs an extensive dataset and utilizes the power of convolutional neural networks for image analysis. It also highlights the feasibility of implementing this approach in clinical settings, offering potential applications in localizing various medical conditions. However, a limitation of this paper is that it focuses primarily on pneumonia detection, and further research is needed to explore its adaptability and performance in the context of other medical conditions or imaging modalities. The study comprehensively employs a sizable dataset and harnesses the capabilities of convolutional neural networks for image analysis. Another limitation of this study is the model's performance and generalizability could be influenced by the dataset's characteristics, potentially leading to overfitting to certain features or biases present in the data. Furthermore, the study does not address the challenges associated with data collection, such as potential variations in image quality, patient demographics, or imaging equipment. Additional research involving a broader range of data sources and more

extensive validation could provide a more comprehensive assessment of the proposed approach's utility in real-world medical settings.

The study by (Tsai and Tao, 2019) focuses on pneumonia detection utilizing the Convolutional Layer, ReLU Layer, and Pooling Layer through a Convolutional Neural Network (CNN) and a decision fusion of feature selection. The algorithm developed consists of a robust 144-layer CNN trained on the extensive Chest X-rays 14 diseases dataset, the largest publicly available collection with over 100,000 frontal view Chest X-rays capturing histological images with 14 diseases. The study successfully detects all 14 diseases in Chest X-rays, showcasing state-of-the-art results across the spectrum. With an average experimental identification rate, the proposed method demonstrates significantly superior performance compared to existing methods, achieving a high classification accuracy rate of 80.90% for the 144 layers of the CNN. The limitation of this study lies in the reliance on a single dataset, Chest X-rays 14 diseases, which, despite its extensive nature, may introduce biases or specific characteristics that limit the generalizability of the proposed algorithm to broader and more diverse datasets. The model's performance may be influenced by the peculiarities of this dataset, potentially leading to overfitting certain features or biases present in the data.

A comprehensive approach for the classification of pneumonia in Chest X-ray images using Convolutional Neural Networks (CNNs) and various transfer learning algorithms in the study by (Narayana Darapaneni et al., 2022). The research explores the use of a non-complex CNN model and integrates transfer learning techniques such as Xception, InceptionV3/V4, and EfficientNetB7 to achieve benchmark results on the Radiological Society of North America (RSNA) dataset. The study addresses the challenge of limited computational resources by experimenting with different methodologies and models. The Mask RCNN model, when applied to a stratified sample of 3017 images with image augmentation, yields a Mean Average Precision (MAP) score of 0.15, showcasing the result. Furthermore, the YoloV3 model, despite achieving a MAP score of 0.32, presents challenges with loss reduction, emphasizing the need for further parameter tuning. However, the limitation of this study lies in the highly unbalanced labels of the RSNA dataset, prompting the use of downsampling techniques, and the overfitting observed in the basic CNN model, which necessitates further exploration of hyperparameters and model architectures to enhance generalization.

The detection of pneumonia in chest radiographs (CXR) by employing an ensemble of deep convolutional neural networks (CNNs), particularly using Mask R-CNN and RetinaNet in the study of (Ko et al., 2019). By focusing on lung opacities as indicators of pneumonia, the study showcases a robust approach to enhance diagnostic accuracy. The dataset, a subset of the NIH dataset from Kaggle, includes 25,684 CXR images with 28,989 lung opacity bounding boxes. The ensemble models, consisting of two Mask R-CNN and three RetinaNet models, demonstrate impressive Mean Average Precision (mAP) values ranging from 0.15035 to 0.21746. Various ensemble strategies, including majority-weighted voting and fine-tuning weight ratios, contribute to the optimization of the detection process. The results highlight that combining multiple CNN architectures(Figure(5)) improves prediction completeness and reduces overall error rates, showcasing the effectiveness of ensemble methods in pneumonia detection. However, a limitation of the study relies on a subset of the NIH dataset, potentially limiting the model's generalization to a broader range of diverse patient populations and imaging conditions. The ensemble methods' success could be influenced by the specific weighting and combination strategies employed, and their robustness across varied datasets and clinical scenarios remains an area for further investigation.

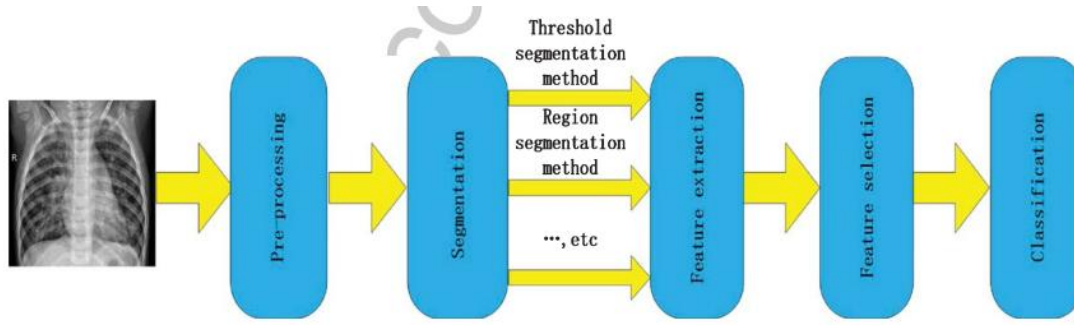


Figure (5):Basic chest X-ray pneumonia detection (Wu et al., 2020)

An approach, of DeepConv-DilatedNet, for identifying and localizing pneumonia in chest X-ray (CXR) images using a two-stage detector Faster R-CNN with a low-complexity dilated bottleneck residual neural network as the backbone. The proposed method in the study of (Yao et al., 2021) combines a Feature Pyramid Network (FPN) with deconvolution for enhanced feature extraction and employs Soft-NMS for anchor box screening, contributing to improved detection accuracy. The study evaluates the algorithm on the Radiological Society of North America (RSNA) dataset, achieving a notable 39.23% Mean Average Precision (mAP), surpassing other detection algorithms. Comparisons with traditional networks like DetNet59, ResNet50, ResNet101, and VGG16 demonstrate the superiority of DeepConv-DilatedNet in pneumonia detection, especially at different IoU thresholds. The main limitation is that the paper lacks an in-depth exploration of potential ethical implications, such as biases in training data or unintended consequences in the clinical decision-making process.

2.7 Research Gap

From the analysis of existing methods, it is evident that pneumonia detection using machine learning and deep learning is capable of learning based on image data. The studies also demonstrate that feature extraction can be effectively employed for detection. However, upon reviewing the literature, it becomes apparent that the majority of the discussed studies used outdated datasets. So in the study performed here a newer and relevant dataset is used.

3 Methodology

3.1 Tools Used

In this research, the study relied on several essential tools, including Python version 3.9, Anaconda, and Visual Studio Code (VS Code). Python 3.9, a versatile and widely adopted programming language, offers a strong foundation for tasks related to data analysis, machine learning, and scientific computing. Anaconda, in contrast, is a well-known open-source distribution platform that streamlines the management of Python environments, packages, and libraries, particularly advantageous for data science and machine learning endeavours. Visual Studio Code, a highly adaptable and lightweight integrated development environment (IDE), played a central role in authoring, editing, and executing Python code with efficiency. The synergy of these tools created a robust and user-friendly environment conducive to data analysis, machine learning, and coding activities, thereby facilitating the exploration, experimentation, and analysis of data for the research study.

3.2 Dataset

The dataset used in this study is a vital component of the research on pneumonia detection from chest X-ray images. This dataset is sourced from the online repository: <https://data.mendeley.com/datasets/rscbjbr9sj/2>. It serves as the foundation for training, validating, and testing machine learning models. The dataset is thoughtfully organized, containing two main folders: one for training data and the other for test data. This division is a standard practice in machine learning, allowing the segregation of the data for model development and evaluation effectively.

Within these folders, the images are further categorized into two fundamental classes: "Normal" and "PNEUMONIA." This binary classification is central to the research objective. The "Normal" class represents chest X-ray images showing no signs of pneumonia, essentially serving as the negative class. In contrast, the "PNEUMONIA" class encompasses images that exhibit characteristics associated with pneumonia, serving as the positive class. This dataset is an invaluable resource for various reasons. Firstly, its real-world clinical relevance cannot be overstated. Chest X-ray images are routinely used in medical diagnostics, making them an ideal choice for medical applications. The presence of both normal and pneumonia-afflicted images ensures a balanced representation of clinical scenarios, mirroring what medical professionals encounter in practice.

The dataset's scale and diversity are notable. With a substantial number of images in both classes, it provides the necessary volume for training robust machine learning models. The diversity within the "PNEUMONIA" class captures the variability in pneumonia presentations, ensuring that models can learn from a wide spectrum of cases. The availability of separate test data is crucial for unbiased model evaluation. The test data is employed to assess how well the models generalize to new, unseen cases. This separation between training and test data mitigates the risk of overfitting, ensuring that models offer reliable predictions when applied to chest X-ray images beyond the dataset.

3.3 Data Preprocessing

Preprocessing and data augmentation are critical stages in developing a deep-learning model for pneumonia detection from chest X-ray images. These processes are instrumental in improving the

model's overall performance and its capacity to generalize effectively. The code provided utilizes the Keras `ImageDataGenerator` for carrying out both preprocessing and data augmentation.

Preprocessing encompasses the preparation of input data to ensure it is appropriately formatted for the deep learning model. In the code, this begins with rescaling the pixel values by dividing them by 255, a normalization procedure that aids in faster convergence during training and mitigates gradient-related challenges. Additionally, the code introduces other operations, including shear transformations, zooming, and horizontal flipping. Shear transformations introduce variability in the data, enhancing the model's adaptability to different angles and perspectives. Zooming simulates varying distances or resolutions in the images, thus enhancing the model's capacity to handle such diversities. Horizontal flipping facilitates improved model generalization by teaching it to recognize pneumonia features on both sides of the chest.

Data augmentation plays a pivotal role in expanding the effective size of the training dataset, an invaluable strategy when working with limited data. Data augmentation contributes to the model's resilience and its ability to generalize across diverse scenarios. It's worth noting that while the provided code applies data augmentation to the validation data, it is typically extended to the training data as well. Common data augmentation techniques encompass rotations, adjustments in width and height, modifications in brightness and contrast, the introduction of blur and noise, scaling, cropping, and elastic deformation. These augmentations introduce diversity into the training data, empowering the model to adapt to potential variations it may encounter in real-world settings.

In the context of pneumonia detection from chest X-ray images, specific data augmentation techniques hold notable value. Techniques such as rotation, width and height adjustments, and horizontal flipping replicate potential variations in X-ray image acquisition. These techniques enable the model to learn to identify pneumonia features from a range of perspectives.

Proficient preprocessing and data augmentation are pivotal in fortifying the deep learning model's accuracy and robustness for pneumonia detection. These steps are especially vital in medical imaging applications, where data quality and variability can substantially influence the model's performance. Meticulous selection and customization of these techniques are essential to align them with the dataset's characteristics and the specific challenges of the task. Effective preprocessing and data augmentation are pivotal in equipping the model to excel in the identification of pneumonia features in chest X-ray images and in extending its learning to real-world medical contexts.

3.4 Training and validation split

The process at hand entails the crucial task of partitioning a dataset into separate training and validation sets, a fundamental step in data preparation for deep learning applications, specifically the detection of pneumonia from chest X-ray images. Batch size, a parameter of paramount importance, governs the number of data samples processed in each training iteration. In this context, a batch size of 32 has been chosen, and its selection significantly influences the trajectory of the training process and the ultimate performance of the model. Smaller batch sizes provide stability and accuracy, albeit potentially demanding more training time. Conversely, larger batch sizes offer computational efficiency but might result in less stable convergence.

The Keras `flow_from_directory` function is leveraged to effectively manage the flow of data, particularly beneficial when dealing with datasets organized into subdirectories, categorized by distinct class labels. This function incorporates several key parameters, such as the specification of directories housing the training and testing data, the definition of target image dimensions (maintaining consistency is pivotal), and the configuration of data shuffling. Data shuffling, when strategically introduced, safeguards against the model learning sequential patterns during training. It's

enabled for the training data generator (`shuffle=True`) while deactivated for the validation data generator (`shuffle=False`) to ensure consistency during evaluation. The primary objectives of data splitting into training and validation subsets are twofold. The training set is dedicated to teaching the model, enabling it to discern patterns and features within the data. Conversely, the validation set plays a pivotal role in assessing the model's ability to generalize and perform effectively when presented with previously unseen data. The training data generator, referred to as `train_generator`, is tasked with producing batches of training data, each batch containing a predefined quantity of images. These batches are systematically processed during the training phase, and data augmentation techniques are judiciously applied to enhance the model's adaptability and resilience. On the other hand, the validation data generator, denoted as `validation_generator`, is responsible for generating data batches sourced from the validation set. This data is instrumental for evaluating the model's performance throughout the training process. It's important to highlight that the validation data remains unshuffled (`shuffle=False`) to ensure consistency and impartiality during the evaluation phase. The process of data splitting and generation forms a foundational pillar of deep learning. It lays the essential groundwork for model training and evaluation, with the choice of parameters and configurations, including batch size, shuffling strategies, and data augmentation, wielding substantial influence over the model's capacity to acquire knowledge and generalize effectively. In the context of this task, the ultimate objective is the accurate detection of pneumonia from chest X-ray images, and the meticulous process delineated here serves as the cornerstone for achieving this goal.

3.5 Deep learning models

Utilizing an ensemble model that integrates InceptionV3, ResNet50, and DenseNet121 for training on the specified dataset represents a sophisticated strategy to elevate model performance. This ensemble approach combines the strengths of multiple deep learning architectures to collectively address the task of detecting pneumonia from chest X-ray images. InceptionV3, ResNet50, and DenseNet121 each contribute their unique capabilities, honed through exposure to diverse datasets, to enhance the overall robustness and accuracy of pneumonia detection. This amalgamation of models resembles consulting multiple experts to arrive at a more precise diagnosis, resulting in a highly efficient model suitable for practical healthcare applications. The collaborative efforts of these models synergize to significantly improve performance and the model's reliability in accurately identifying pneumonia cases in chest X-ray images.

Inception V3

The model training process for pneumonia detection from chest X-ray images is a highly structured and sophisticated procedure. It leverages the robust InceptionV3 as the foundational model. InceptionV3's pre-training on the ImageNet dataset, an extensive collection of images spanning numerous categories, equips it with a wealth of knowledge in feature extraction and pattern recognition. The model's architecture is meticulously customized to cater to the binary classification task at hand, commencing with Global Average Pooling and progressing through dense layers with ReLU activation functions. The final layer employs the sigmoid activation function to provide probability scores indicating the likelihood of pneumonia presence. When it comes to compiling the model, the selection of the 'Adam' optimizer, 'binary_crossentropy' loss function, and accuracy metric is crucial. The training process involves transfer learning, initially freezing InceptionV3's layers to retain essential features.

The training phase entails exposing the model to labeled chest X-ray images and iteratively adjusting its parameters to minimize the binary cross-entropy loss while enhancing accuracy. Validation data plays a pivotal role in assessing the model's performance during training. This process ensures the

model's ability to generalize to new, unseen data and mitigates the risk of overfitting. Subsequently, the fine-tuning stage, initiated after the initial training, further enhances the model's capability by selectively unfreezing some InceptionV3 layers to adapt to the nuances of the pneumonia detection task. Model training is an iterative journey that necessitates adjustments to hyperparameters and training strategies to optimize results. The result is a finely tuned deep learning model ready for real-world pneumonia detection, exemplifying the synergy between advanced machine learning techniques and their application in healthcare.

At the core of this pneumonia detection model is the state-of-the-art convolutional neural network, InceptionV3. Its exceptional strength is rooted in its pre-trained status on the ImageNet dataset, which spans a vast spectrum of image categories. InceptionV3 has effectively learned to extract intricate features from this diverse image corpus, encompassing everything from basic shapes and edges to complex textures and object parts. This extensive knowledge, embodied in InceptionV3's architecture and weights, becomes invaluable for the task of pneumonia detection from chest X-ray images. Incorporating InceptionV3 as a foundation, the model is thoughtfully customized to align with the specific requirements of pneumonia detection. The customization process unfolds with several critical steps:

Following feature extraction by InceptionV3, a Global Average Pooling 2D layer is introduced. This layer operates on the feature maps produced by InceptionV3 and compresses the spatial information into a single vector. The outcome simplifies the data while retaining vital features. This not only streamlines computational complexity but also concentrates on essential information. Beyond global average pooling, the model is equipped with two dense (fully connected) layers. The first dense layer boasts 1024 units and utilizes the Rectified Linear Unit (ReLU) activation function. ReLU's introduction of non-linearity empowers the model to capture intricate data patterns. These dense layers play a pivotal role in making high-level predictions grounded in the features extracted by InceptionV3. The final layer comprises a single unit, utilizing the sigmoid activation function. This configuration aligns with the binary classification task, which requires discerning chest X-ray images indicating pneumonia from those that do not. The sigmoid activation function condenses the model's output into a probability score, thereby indicating the likelihood of an image containing pneumonia. This score informs the final classification.

Following the architecture's customization, the model undergoes the compilation phase, which involves making several critical decisions. The chosen optimizer is 'Adam.' This selection is notable for the algorithm's adaptive learning rate, which adjusts during training to facilitate efficient convergence. Such adaptability accelerates the model's journey to find optimal solutions. For this binary classification task, the loss function is 'binary_crossentropy.' This choice is well-suited to situations where the model's objective is to classify each image as either pneumonia (positive class) or non-pneumonia (negative class). Binary cross-entropy quantifies the disparity between predicted probabilities and actual binary labels, guiding the model toward improved classifications. The model's performance is evaluated using the accuracy metric, which measures how effectively the model predicts the correct labels, signifying the number of accurately classified images. While accuracy provides a broad view of performance, specialized applications may warrant consideration of additional metrics. One of the foundational principles in this model's training strategy is transfer learning. Transfer learning involves leveraging a pre-trained model, in this case, InceptionV3, as the starting point for the pneumonia detection task. In the initial training stages, InceptionV3's layers remain frozen, preserving their weights and parameters. This strategy aims to safeguard the wealth of valuable features learned from the extensive ImageNet dataset.

With the customized architecture, compiled model, and the bedrock of transfer learning in place, the model embarks on the training phase. The model is exposed to a substantial dataset of labeled chest X-ray images, where each image is meticulously annotated to indicate the presence or absence of

pneumonia. The model iteratively adjusts its internal parameters to minimize the binary cross-entropy loss and enhance accuracy. This iterative process forms the core of deep learning, where the model progressively learns to recognize patterns and features indicative of pneumonia in X-ray images.

ResNet-50

The comprehensive model training process for pneumonia detection from chest X-ray images using the ResNet50 architecture as the foundational model. This detailed walkthrough demonstrates the intricate steps involved in crafting an accurate and reliable model for real-world healthcare applications. The core of our model is the formidable ResNet50 architecture, renowned for its capabilities in various computer vision tasks. Its exceptional strength stems from its pre-training on the ImageNet dataset, a vast collection of images covering diverse categories. Leveraging this pre-trained model equips our model with a deep understanding of feature extraction and pattern recognition. In the context of pneumonia detection, this knowledge is invaluable, as it allows our model to discern critical features in chest X-ray images indicative of pneumonia. While ResNet50 provides a robust foundation, customization is essential to tailor the architecture to the specific demands of pneumonia detection. This customization journey involves key components: Following the feature extraction phase, we introduce a Global Average Pooling 2D layer. This layer operates on the feature maps generated by ResNet50, summarizing the spatial information into a single vector. This simplification not only reduces computational complexity but also emphasizes essential features. Beyond global average pooling, two dense (fully connected) layers are incorporated. The first dense layer boasts 1024 units and employs the Rectified Linear Unit (ReLU) activation function, introducing non-linearity and enabling the model to capture intricate patterns within the data. The final layer is a dense layer housing a single unit, utilizing the sigmoid activation function. This configuration is tailored to the binary classification task of discerning between chest X-ray images indicating pneumonia and those that do not. The sigmoid activation function transforms the model's output into a probability score, offering insight into the likelihood of an image containing pneumonia.

Compiling the model is a pivotal step, involving the selection of the optimizer, loss function, and evaluation metrics. The 'Adam' optimizer is chosen for its efficiency and adaptability in adjusting the learning rate during training. This adaptability expedites convergence, helping the model to find optimal solutions effectively. For this binary classification task, 'binary_crossentropy' serves as the loss function. This choice is apt for situations where the model's primary objective is to categorize images as either pneumonia-positive (belonging to the positive class) or pneumonia-negative (belonging to the negative class). Binary cross-entropy quantifies the disparity between predicted probabilities and actual binary labels, guiding the model to make more accurate classifications. To gauge the model's performance, the accuracy metric is employed, providing a broad overview of its effectiveness in predicting the correct labels. Depending on the specific application, additional metrics might be considered for a more comprehensive evaluation. An indispensable aspect of this model's training strategy is transfer learning. This concept revolves around the utilization of a pre-trained model, in this case, ResNet50, as the foundation for the pneumonia detection task. In the initial training stages, the layers of ResNet50 are frozen, preserving their learned weights and parameters. This strategy aims to retain the wealth of valuable features and knowledge accumulated during the model's training on the extensive ImageNet dataset.

With the customized architecture, compiled model, and the principles of transfer learning in place, the model is primed for the training phase. During this phase, the model encounters a substantial dataset of chest X-ray images, each meticulously labeled to denote the presence or absence of pneumonia. The model iteratively adapts its internal parameters to minimize the binary cross-entropy loss and enhance its accuracy. This iterative learning process is the essence of deep learning, allowing the model to gradually identify patterns and features that indicate the presence of pneumonia in X-

ray images. Following the initial training phase, the opportunity for fine-tuning arises. Fine-tuning involves selectively unfreezing certain layers of the ResNet50 base model, allowing them to adapt to the specific nuances of the pneumonia detection task. This fine-tuning phase can further enhance the model's performance by tailoring it to the unique characteristics of the dataset. Model training is often an iterative journey, involving the continuous adjustment of hyperparameters and the exploration of different training strategies to optimize results. Regular monitoring, validation, and fine-tuning are fundamental to achieving the best possible model performance. The model training process for pneumonia detection from chest X-ray images is a meticulously structured and iteratively refined journey. By incorporating ResNet50 as the foundational architecture, and through the thoughtful customization of the model's components, it becomes finely attuned to the specific

DenseNet-121

In this journey, we will focus on the utilization of the DenseNet121 architecture as the foundational model. This comprehensive explanation will highlight the critical steps and considerations in crafting a high-performing model for real-world healthcare applications.

At the heart of our pneumonia detection model lies the robust DenseNet121 architecture, celebrated for its exceptional performance in computer vision tasks. Its strength is fortified by its pre-training on the ImageNet dataset, a massive repository of images spanning numerous categories. This extensive pre-training equips our model with an innate understanding of feature extraction and pattern recognition. In the context of pneumonia detection, this knowledge is invaluable, as it allows the model to discern crucial features in chest X-ray images that signify the presence of pneumonia. While DenseNet121 provides a strong foundation, customization is essential to adapt the architecture to the specific requisites of pneumonia detection. This customization process involves several pivotal components: Following the feature extraction phase by DenseNet121, we introduce a Global Average Pooling 2D layer. This layer acts on the feature maps generated by DenseNet121, condensing the spatial information into a single vector. This simplification streamlines computational complexity and accentuates essential features. Beyond global average pooling, we incorporate two dense (fully connected) layers. The first dense layer comprises 1024 units and utilizes the Rectified Linear Unit (ReLU) activation function. Introducing non-linearity via ReLU empowers the model to capture intricate data patterns. The final layer is a dense layer housing a single unit and employing the sigmoid activation function. This configuration is meticulously tailored to the binary classification task, enabling the distinction between chest X-ray images indicating pneumonia and those that do not. The sigmoid activation function compresses the model's output into a probability score, providing insight into the likelihood of an image containing pneumonia.

The compilation of the model is a pivotal step, encompassing decisions related to the optimizer, loss function, and evaluation metrics. For this model, we opt for the 'Adam' optimizer. Adam stands out for its efficiency and adaptability in dynamically adjusting the learning rate during training. This adaptability expedites the convergence process, facilitating the model in finding optimal solutions efficiently. In binary classification, 'binary_crossentropy' serves as the chosen loss function. This choice is well-suited for scenarios where the model's primary goal is to categorize images as either pneumonia-positive (belonging to the positive class) or pneumonia-negative (belonging to the negative class). Binary cross-entropy quantifies the disparity between predicted probabilities and actual binary labels, guiding the model to make more accurate classifications. To gauge the model's performance, we rely on the accuracy metric. Accuracy provides a broad overview of the model's effectiveness in correctly predicting labels. However, it's important to note that specialized applications may warrant the consideration of additional metrics for a more comprehensive evaluation. A fundamental principle underpinning our model's training strategy is transfer learning. Transfer learning involves harnessing a pre-trained model, in this case, DenseNet121, as the foundation for the pneumonia detection task. During the initial stages of training, the layers of

DenseNet121 are frozen, preserving their pre-learned weights and parameters. This strategy is pivotal in safeguarding the valuable features and knowledge accrued during DenseNet121's training on the extensive ImageNet dataset.

Armed with the customized architecture, the compiled model, and the principles of transfer learning, our model is primed for the training phase. During this phase, the model is exposed to an extensive dataset of labeled chest X-ray images. Each image is meticulously annotated to indicate the presence or absence of pneumonia. The model, through iterative adjustments, aims to minimize the binary cross-entropy loss while enhancing accuracy. This iterative learning process is at the core of deep learning, enabling the model to progressively recognize patterns and features indicative of pneumonia in X-ray images. A pivotal component of the model training process is evaluation. During the training process, the model's performance is rigorously assessed using a separate dataset known as the validation set. This dataset comprises chest X-ray images distinct from the training data and serves as a benchmark for evaluating the model's generalization capabilities. Evaluating the model on this independent dataset ensures that it doesn't merely memorize the training data, a phenomenon known as overfitting, and that it can accurately classify new and unseen images. The model's performance is closely monitored in terms of accuracy and loss on this validation dataset, offering insights into its robustness and generalization capabilities. Following the initial training phase, the opportunity for fine-tuning emerges. Fine-tuning involves selectively unfreezing specific layers of the DenseNet121 base model, allowing them to adapt to the specific nuances of the pneumonia detection task. This fine-tuning phase can further enhance the model's performance by tailoring it to the unique characteristics of the dataset. Model training is often an iterative journey. It involves the continuous adjustment of hyperparameters and the exploration of different training strategies to optimize results. Regular monitoring, validation, and fine-tuning are fundamental to achieving the best possible model performance. The model training process for pneumonia detection from chest X-ray images is a meticulously structured and iteratively refined journey. By incorporating DenseNet121 as the foundational architecture and through the thoughtful customization of the model's components, it becomes finely attuned to the specific requirements of this critical healthcare application. The blend of deep learning techniques, transfer learning, and diligent training protocols is fundamental to the success of this model, ultimately benefiting the field of medical imaging and patient care.

3.6 Performance Evaluation

Following the model training phase, a comprehensive performance assessment is carried out using a range of critical metrics and tools. These encompass accuracy, which gauges the overall correctness of the model's predictions; recall, which evaluates the model's capacity to identify cases of pneumonia accurately; precision, quantifying the model's precision when predicting pneumonia; the F1 score, which balances precision and recall; and the confusion matrix, offering an intricate breakdown of true positives, true negatives, false positives, and false negatives. Additionally, the support metric provides context by indicating the number of instances within each class. This combined set of metrics enables a meticulous evaluation of the model's efficacy in pneumonia detection from X-ray images, supplying invaluable insights for your report concerning its real-world utility and dependability.

Accuracy

When assessing the performance of machine learning and deep learning models, the metric of "accuracy" takes centre stage. Accuracy, as a performance evaluation measure, quantifies the model's capacity to make correct predictions within a given dataset. This metric is computed by dividing the number of instances that were predicted accurately by the model by the total number of instances in

the dataset. Essentially, accuracy offers a lucid and straightforward perspective on the model's overall correctness. It serves as a fundamental benchmark for evaluating the model's performance, offering valuable insights. Nonetheless, it's worth recognizing that accuracy may not be the exclusive yardstick of performance, particularly in situations marked by imbalanced datasets or when different types of errors hold varying significance.

While accuracy is a straightforward and easily understandable metric, it may not always be the most appropriate option for performance evaluation. This is especially true in cases where the dataset demonstrates an imbalance, where one class significantly outweighs the others, or when different error types have varying consequences.

Recall

Recall, sometimes known as sensitivity or the true positive rate, is a performance assessment metric commonly used in classification tasks. Its principal objective is to measure the model's ability to correctly identify all relevant instances of a specific class within a provided dataset. In the realm of classification tasks, Recall, also known as sensitivity or the true positive rate, stands as a vital performance metric. Its primary aim is to assess the model's capacity to accurately detect all pertinent instances of a particular class within a given dataset.

Precision

Precision, when employed as a performance assessment metric within classification tasks, emphasizes the model's ability to generate accurate positive predictions, particularly when the aim is to reduce false positives. Precision directs its focus toward the accuracy of the model's positive predictions. It measures the fraction of positive predictions that are genuinely correct, and this is particularly vital when the presence of false positives could result in substantial adverse consequences.

F1 Score

The F1 score is a performance evaluation metric used in classification scenarios, and it combines precision and recall into a single composite value. It becomes particularly useful when there is a requirement to find a balance between precision and recall, essentially seeking an equilibrium between making precise positive predictions and correctly identifying all pertinent instances of the positive class. The F1 score, which unifies precision and recall, provides a balanced assessment that takes into account the precision of positive predictions and the model's capability to correctly identify all pertinent positive instances. This metric proves especially valuable in scenarios involving imbalanced class distributions or when the implications of false positives and false negatives differ significantly.

Confusion Matrix

The confusion matrix is a valuable instrument for evaluating the effectiveness of machine learning models, especially in classification tasks. This matrix functions as a tabular representation that concisely summarizes the model's predictions and their agreement or variance concerning the true class labels in a dataset. Typically, a confusion matrix includes four key values:

- **True Positives (TP):** These are instances that the model correctly predicted as belonging to the positive class.

- True Negatives (TN): These represent instances that the model correctly predicted as belonging to the negative class.
- False Positives (FP): These are instances that the model incorrectly predicted as positive when, in reality, they are negative.
- False Negatives (FN): These denote instances that the model incorrectly predicted as negative when, in fact, they are positive.

The significance of the confusion matrix lies in its capacity to offer a detailed dissection of the model's performance.

4 Results

4.1 Accuracy

The accuracies of DenseNet-121, ResNet-50, and InceptionV3 were determined. The accuracy values of the four models are displayed below in table(1) and

Models	Accuracy(in %)
DenseNet121	91 %
Inception V3	85 %
Resnet50	73 %

Table (1): Comparison of accuracy

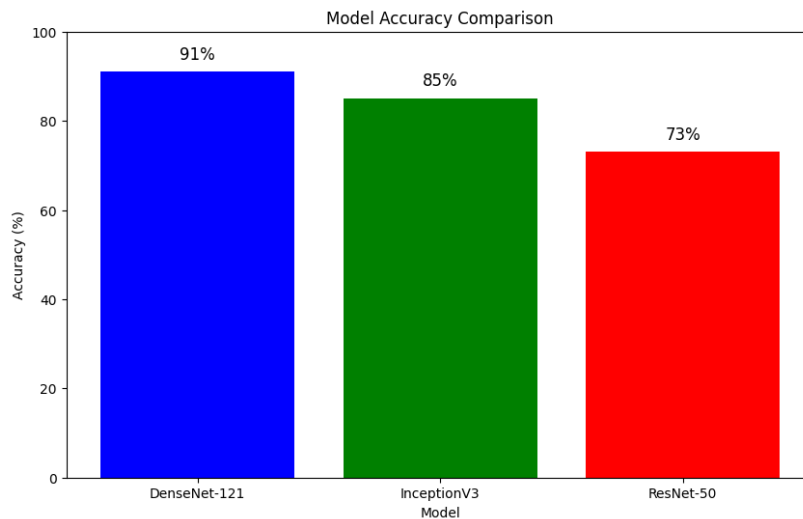


Figure (6): comparison of accuracies

4.2 Precision

The precision of DenseNet-121, ResNet-50, and InceptionV3 were determined. The precision values of the four models are displayed below in table(2).

Models	Precision (in %)
DenseNet121	91 %
Inception V3	87 %
Resnet50	76 %

Table (2): Precision values

4.3 Recall

The Recall of DenseNet-121, ResNet-50, and InceptionV3 were determined. The recall values of the four models are displayed below.

Models	Recall (in %)
DenseNet121	91 %
Inception V3	85 %
Resnet50	73 %

Table (3): Recall values of the models

4.4 F1 Score

The F1 score of DenseNet-121, ResNet-50, and InceptionV3 were determined. The F1 score values of the four models are displayed below.

Models	F1 Score (in %)
DenseNet121	91 %
Inception V3	84 %
Resnet50	70 %

Table(4): F1-score values

4.5 Confusion Matrix

The confusion matrices generated by the DenseNet-121, InceptionV3, and ResNet-50, models are given in figures (7), (8), and (9), respectively.

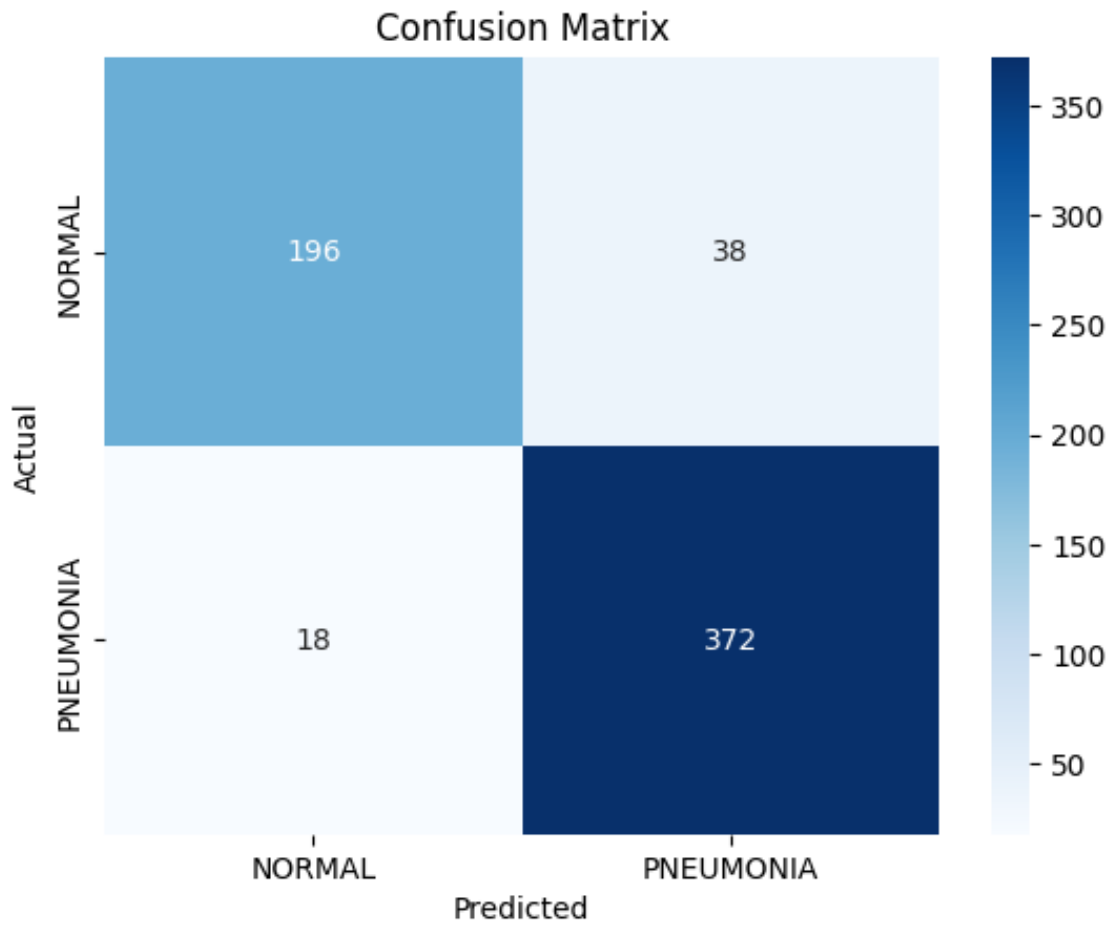


Figure (7): Confusion matrix of DenseNet-121

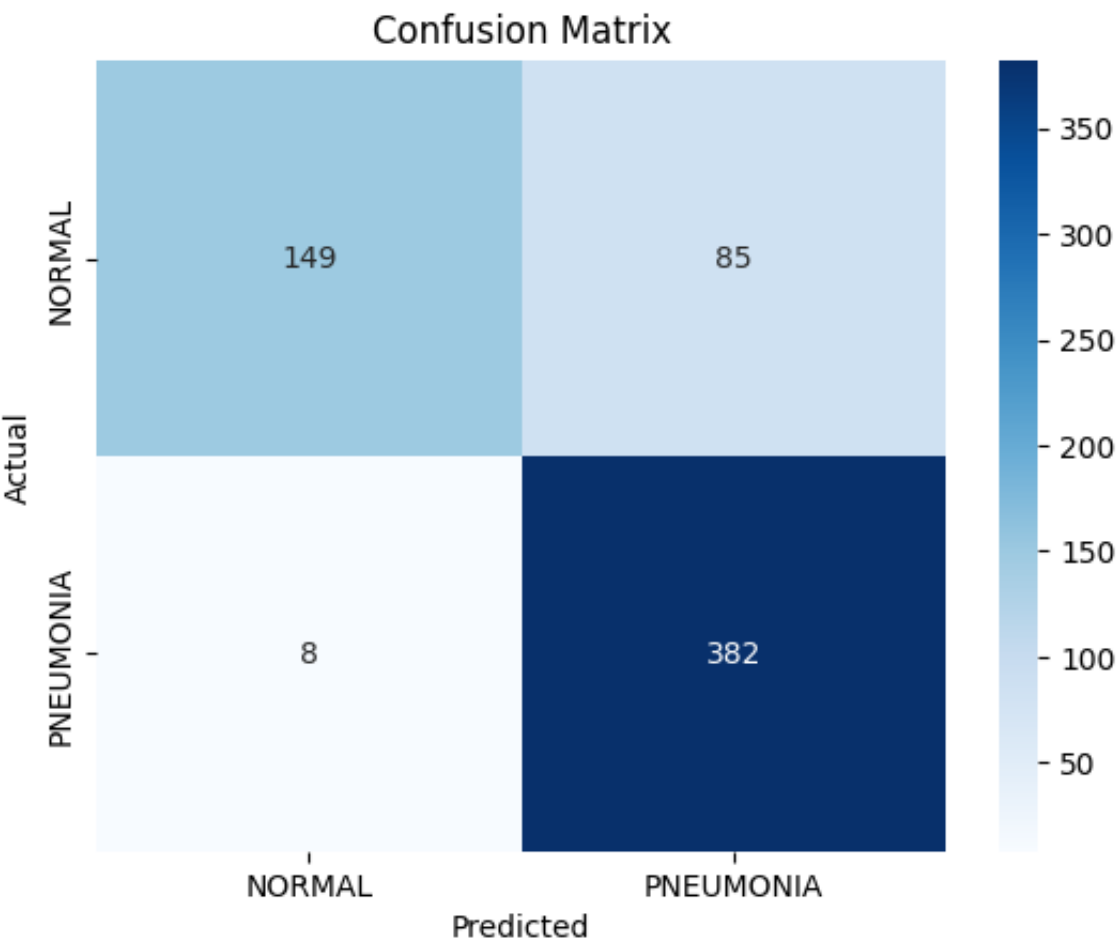


Figure (8): Confusion matrix of Inception-V3

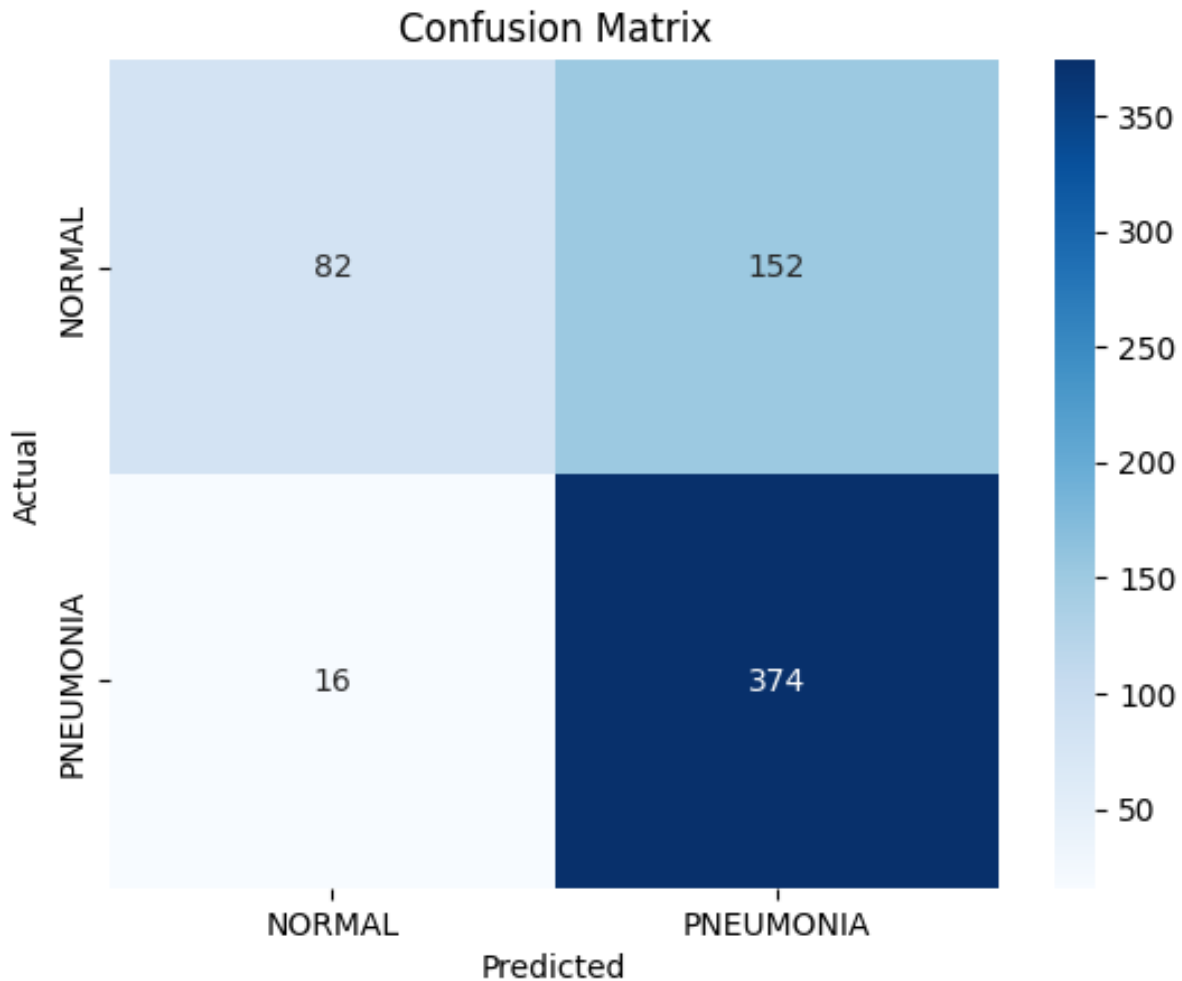


Figure (9): Confusion matrix Resnet-50

4.6 Graphical user interface (GUI)

A Graphical User Interface (GUI) has been developed using the Tkinter library for the ensemble model consisting of InceptionV3, ResNet50, and DenseNet121. When you launch the desktop application, you will encounter two buttons: "Upload Image" and "Predict."

Upon clicking the "Upload Image" button, you can select a test image from your local files. After successfully uploading the image, a message box will appear, confirming that the image has been selected. Next, you proceed by clicking the "Predict" button. Once this button is activated, the ensemble model will analyse the uploaded image. Following this analysis, a message box will pop up, displaying the prediction result. The prediction will indicate whether the image shows signs of pneumonia (PNEUMONIA detected) or if it is a normal chest X-ray (normal)(Figure(10)).

This user-friendly interface simplifies the process of testing the pneumonia detection model on your images, providing immediate feedback on the model's classification results. This desktop application facilitates convenient and quick predictions for real-world medical image analysis.

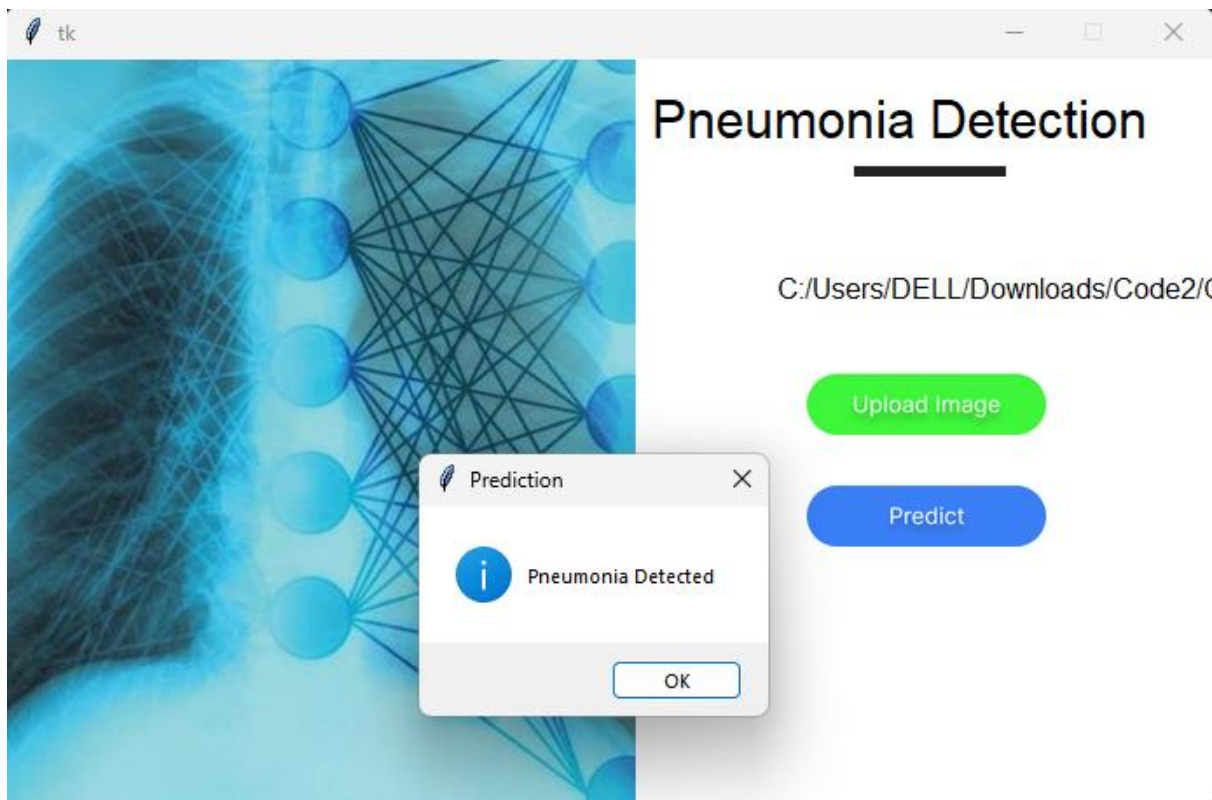


Figure (10): GUI

5 Project Management

5.1 Project Schedule

The project aimed to achieve its primary goal by creating a detailed project schedule, as illustrated in a Gantt chart from 12th October 2023 to 30th November 2023. This chart was divided into several phases, starting with preliminary research tasks on 12th October 2023, followed by image preprocessing on 19th October 2023, data augmentation on 24th October 2023, ensemble deep learning model creation on 30th October 2023, model training from 10th November 2023, prediction tasks starting from 20th November 2023, and finally, the GUI integration starts from 24th November 2023. This structured timeline aimed to ensure a methodical approach to project execution.

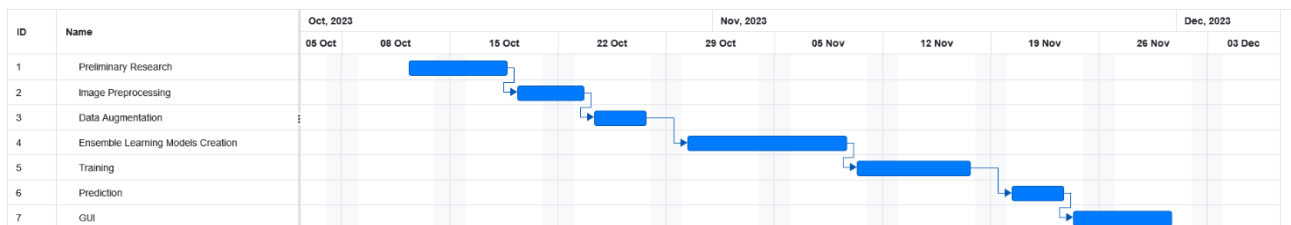


Figure (11): Gantt Chart

After reviewing the chart in figure(11) it's clear that each task had specific start and end dates, treated as actual deadlines with sincere efforts to follow the schedule. However, the ensemble model training took longer than originally planned. But the time was effectively managed to accommodate these extended durations.

5.2 Risk Management

Identifying potential risks and finding suitable solutions is crucial for successful project management, especially in the face of unexpected events. In this project, we have outlined various risks and their corresponding solutions in the provided table(5).

No.	Risk	Mitigation	Solution	Impact
1	Less quantity of images	Regularly analyse the total image dataset during preprocessing.	Performed data augmentation techniques to increase the volume and diversity of the available images.	Medium
2	The issue with technical aspects related to image processing	Implement hardware acceleration, such	Employed cloud environments, like Google	Medium

		as GPU utilization, to expedite image processing tasks.	Colab, to leverage complimentary GPU resources for accelerated computations.	
--	--	---	--	--

Table(5): Risk management

It can be concluded that proactive measures were taken to mitigate identified risks. Regular dataset analysis and data augmentation were implemented to address the risk of a smaller quantity of images, while hardware acceleration and the use of cloud environments with complimentary GPU access were employed to mitigate issues related to technical aspects of image processing. These actions were taken to manage the potential medium impact associated with these risks effectively.

5.3 Quality Management

To maintain project progress, a key strategy involved conducting weekly meetings with the supervisor to receive feedback on completed tasks and discuss upcoming steps. Additionally, we regularly updated the Gantt chart and utilized a digital calendar for timely deadline reminders.

5.4 Social, Legal, Ethical and Professional Considerations

Building a deep learning mechanism for pneumonia detection from chest X-ray images requires a specific approach, taking into account various social, legal, ethical, and professional considerations. The image dataset employed in this study is treated with the utmost ethical consideration, ensuring the exclusion of sensitive information to uphold privacy and adhere to ethical standards. Legal compliance is underscored through strict adherence to copyright laws, and the construction of the deep learning models (DenseNet-121, ResNet-50, Inception-V3) involves building the codebase from the ground up, reinforcing originality and respect for intellectual property rights. From a social standpoint, the Python Tkinter GUI is carefully crafted to be inclusive and non-discriminatory, avoiding offensive symbols or designs that may be discriminatory to any specific group or community. This socially responsible approach ensures that the technology developed is not only advanced in functionality but also considerate of diverse user groups. Professionalism is a guiding principle throughout the project's lifecycle, with a well-structured plan governing the training, testing, and evaluation of the models, thus mitigating last-minute pressures and stress. The transparent presentation of performance evaluation metrics and the secure storage of high-accuracy models for future use showcase a commitment to accountability and reliability in deploying the pneumonia detection mechanism. Ethical considerations extend into the prediction phase, where careful preprocessing of input images is conducted to meet the model's requirements. Trained models with high accuracy are chosen based on performance metrics, maintaining a dedication to delivering trustworthy pneumonia predictions. The outcomes are showcased in the GUI, contributing to the responsible and ethical application of deep learning technology in pneumonia detection from chest X-ray images.

6 Discussion and evaluation

6.1 Discussion

The study successfully achieved its main goals. The primary aim was to develop a pneumonia detection system using an ensemble of deep learning models. The specific objectives were:

Transform the dataset images into a format compatible with training the deep learning models

The objective was achieved successfully, and the dataset was successfully transformed into a format compatible with training through data preprocessing and data augmentation.

Determine the performance metrics of the three deep learning models used here, separately

The objective was achieved successfully, and the performance metrics including confusion matrix, precision, recall, F1 score, test accuracy, as well as visualizations for validation accuracy and validation loss were successfully determined for each of the three deep learning models: ResNet50, DenseNet121, and InceptionV3.

Determine the deep learning model which has the best performance in pneumonia detection.

The objective was achieved successfully, and it was found that DenseNet121 exhibited the best performance in pneumonia detection, with a test accuracy of 91.03%.

Create a desktop application that identifies pneumonia based on an X-ray image given as input.

The objective was achieved successfully, and a user-friendly desktop application was created. The application allows users to input X-ray images and view the pneumonia detection results through a message box.

Compare the performance of the best-performing deep learning model used here to the existing deep learning-based pneumonia detection systems.

The best-performing deep learning model's performance was compared to other existing models for predicting pneumonia in X-ray images. This comparison included metrics like accuracy, to thoroughly assess the model's effectiveness.

Accuracy was chosen for measuring the models' effectiveness in classifying the classes correctly. Table (6) provides a comparative summary of pneumonia detection accuracies achieved by both the existing systems and the system developed in this study.

Model	Accuracy(in %)
Kumar et al.(2023)	93.12%
Md Shad Akhtar et al.(2016)	91%
Tsai and Tao(2019)	80.90%

Al Mamlook, Chen and Bzizi(2020)	98.46%
Barhoom, Samy and Naser(2022)	98.2%
Racic et al.(2021)	88.9%
El Asnaoui, Chawki and Idri(2021)	96%
Puneet Gupta(2021)	93%
Proposed System	91.03%

Table (6): the accuracies of the existing systems with the system built here

6.2 Critical evaluation

The project primarily focuses on a selected set of models, and the exclusion of other emerging architectures or advanced algorithms might limit the exploration of potential improvements in detection accuracy. Moreover, the reliance on a specific dataset for training and evaluation introduces the risk of dataset bias, and the generalizability of the models to diverse populations or imaging conditions may be compromised.

7 Conclusion and future enhancements

7.1 Conclusion

The pneumonia detection system using ensemble deep learning models was successfully developed in this project. Utilizing architectures such as DenseNet121, InceptionV3, and ResNet50, the models were trained and evaluated using a publicly available chest X-ray dataset sourced from Mendeley Data. Pre-processing and data augmentation played a crucial role in enhancing the efficiency of the models. The training phase involved model compilation, utilization of the training dataset, and performance assessment, with the trained models being saved for future use. During the prediction phase, the three models effectively detected pneumonia in the test data, showcasing their effectiveness in X-ray image analysis. The study comprehensively addressed and provided solutions to the primary research questions:

- Which deep learning model among the DenseNet-121, ResNet-50, and Inception-V3 will have the better performance in detecting pneumonia from X-ray images?

DenseNet121 exhibited the best performance in detecting pneumonia from X-ray images, with a test accuracy of 91.03%, outperforming ResNet50 and InceptionV3.

- Will the performance of the best-performing model in this study in detecting pneumonia be greater than the performance of an existing deep learning-based pneumonia detection system?

The suggested approach utilizes ensemble learning, incorporating DenseNet121, ResNet50, and InceptionV3 models. Notably, DenseNet121 achieved a test accuracy of 91.03%, which was lower than the accuracy of an existing system. It's essential to acknowledge that certain other systems might exhibit higher accuracy, but variations in processes and datasets can account for these differences.

7.2 Future enhancements

As we move forward, there are several methods for enhancing the capabilities of our pneumonia detection system. Firstly, exploring the integration of transfer learning techniques could significantly improve model performance. Transfer learning involves leveraging pre-trained models on large datasets and fine-tuning them for our specific task, potentially enhancing the model's ability to extract relevant features from chest X-ray images. Furthermore, investigating the

deployment of advanced architectures or ensembles with attention mechanisms and interpretability features could provide insights into the decision-making process of the model, fostering better trust and understanding of the predictions.

8 Student Reflections

Creating a deep learning system to detect pneumonia from chest X-ray images has been a challenging yet rewarding journey. Dealing with tasks like image preprocessing, ensemble learning, and using advanced models like DenseNet-121, ResNet-50, and Inception-V3 seemed complex at first, but I learned a lot. The skills I gained are not just limited to pneumonia detection; they provide a strong foundation for other machine learning and data science projects. Despite facing challenges, I successfully completed the project by sticking to a well-organized plan and keeping open communication with my mentors, whose guidance played a crucial role in the project's success. This experience not only improved my technical skills but also boosted my confidence in solving complex problems, setting the stage for further growth in the field.

References

- Al Mamlook, R.E., Chen, S. and Bzizi, H.F. (2020). Investigation of the performance of Machine Learning Classifiers for Pneumonia Detection in Chest X-ray Images. 2020 IEEE International Conference on Electro Information Technology (EIT). doi:<https://doi.org/10.1109/eit48999.2020.9208232>.
- Albelwi, S.A. (2022). Deep Architecture based on DenseNet-121 Model for Weather Image Recognition. *International Journal of Advanced Computer Science and Applications*, 13(10). doi:<https://doi.org/10.14569/ijacsa.2022.0131065>.
- Ali, N., Quansah, E., Köhler, K., Meyer, T., Schmitt, M., Popp, J., Niendorf, A. and Bocklitz, T. (2019). Automatic label-free detection of breast cancer using nonlinear multimodal imaging and the convolutional neural network ResNet50. *Translational Biophotonics*, 1(1-2). doi:<https://doi.org/10.1002/tbio.201900003>.
- Barhoom, A., Samy, P. and Naser, S. (2022). Diagnosis of Pneumonia Using Deep Learning. *International Journal of Academic Engineering Research (IJAER)*, [online] 6, pp.48–68. Available at: <https://philarchive.org/archive/BARDOP-3> [Accessed 25 Nov. 2023].
- Cao, J., Yan, M., Jia, Y., Tian, X. and Zhang, Z. (2021). Application of a modified Inception-v3 model in the dynasty-based classification of ancient murals. *EURASIP Journal on Advances in Signal Processing*, 2021(1). doi:<https://doi.org/10.1186/s13634-021-00740-8>.
- Effah, C.Y., Miao, R., Drokow, E.K., Agboyibor, C., Qiao, R., Wu, Y., Miao, L. and Wang, Y. (2022). Machine learning-assisted prediction of pneumonia based on non-invasive measures. *Frontiers in Public Health*, 10. doi:<https://doi.org/10.3389/fpubh.2022.938801>.
- El Asnaoui, K., Chawki, Y. and Idri, A. (2021). Automated Methods for Detection and Classification Pneumonia Based on X-Ray Images Using Deep Learning. *Studies in Big Data*, pp.257–284. doi:https://doi.org/10.1007/978-3-030-74575-2_14.
- Franquet, T. (2018). Imaging of Community-acquired Pneumonia. *Journal of Thoracic Imaging*, 33(5), pp.282–294. doi:<https://doi.org/10.1097/rti.0000000000000347>.
- Gabruseva, T., Poplavskiy, D. and Kalinin, A. (2020). Deep Learning for Automatic Pneumonia Detection. [online] Available at: https://openaccess.thecvf.com/content_CVPRW_2020/papers/w22/Gabruseva_Deep_Learning_for_Automatic_Pneumonia_Detecti
- Guan, Q., Wan, X., Lu, H., Ping, B., Li, D., Wang, L., Zhu, Y., Wang, Y. and Xiang, J. (2019). Deep convolutional neural network Inception-v3 model for differential diagnosing of lymph node in cytological images: a pilot study. *Annals of Translational Medicine*, 7(14), pp.307–307. doi:<https://doi.org/10.21037/atm.2019.06.29>.

- Hamlili, F.-Z., Beladgham, M., Khelifi, M. and Boudia, A. (2022). Transfer learning with Resnet-50 for detecting COVID-19 in chest X-ray images. *Indonesian Journal of Electrical Engineering and Computer Science*, 25(3), p.1458. doi:<https://doi.org/10.11591/ijeecs.v25.i3.pp1458-1468>.
- Islam, S.R., Maity, S.P., Ray, A.K. and Mandal, M. (2019). Automatic Detection of Pneumonia on Compressed Sensing Images using Deep Learning. [online] *IEEE Xplore*. doi:<https://doi.org/10.1109/CCECE.2019.8861969>.
- Jain, R., Nagrath, P., Kataria, G., Sirish Kaushik, V. and Jude Hemanth, D. (2020). Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning. *Measurement*, 165, p.108046. doi:<https://doi.org/10.1016/j.measurement.2020.108046>.
- Kelly, B. (2012). The chest radiograph. *The Ulster medical journal*, [online] 81(3), pp.143–8. Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3632825/>.
- Khan, W., Zaki, N. and Ali, L. (2021). Intelligent Pneumonia Identification From Chest X-Rays: A Systematic Literature Review. *IEEE Access*, 9, pp.51747–51771. doi:<https://doi.org/10.1109/access.2021.3069937>.
- Kirandeep and Kirandeep, Ramanpreet Kaur, Vijay Dhir(2023). IMAGE RECOGNITION USING RESNET50. [online] Available at:<https://www.eurchembull.com/uploads/paper/3d3038bc3e7d1fad63f18fa13fe711ac.pdf>
- Ko, H., Ha, H., Cho, H., Seo, K. and Lee, J. (2019). Pneumonia Detection with Weighted Voting Ensemble of CNN Models. doi:<https://doi.org/10.1109/icaibd.2019.8837042>.
- Kumar, P.P. (2023). Pneumonia Detection Using Deep Learning Methods. *International Journal for Research in Applied Science and Engineering Technology*, [online] 11(4), pp.1789–1793. doi:<https://doi.org/10.22214/ijraset.2023.50483>.
- Liang, J. (2020). Image classification based on RESNET. *Journal of Physics: Conference Series*, 1634, p.012110. doi:<https://doi.org/10.1088/1742-6596/1634/1/012110>.
- Manoj krishna, M., Neelima, M., Harshali, M. and Venu Gopala Rao, M. (2018). Image classification using Deep learning. *International Journal of Engineering & Technology*, 7(2.7), p.614. doi:<https://doi.org/10.14419/ijet.v7i2.7.1082>.
- Mukherjee, S. (2022). The Annotated ResNet-50. [online] *Medium*. Available at: <https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758>.
- Narayana Darapaneni, Ranjan, A., Bright, D., Trivedi, D., Kumar, K., Kumar, V. and Anwesh Reddy Paduri (2022). Pneumonia Detection in Chest X-Rays using Neural Networks. *arXiv (Cornell University)*. doi:<https://doi.org/10.48550/arxiv.2204.03618>.
- Neuman, M.I., Lee, E.Y., Bixby, S., Diperna, S., Hellinger, J., Markowitz, R., Servaes, S., Monuteaux, M.C. and Shah, S.S. (2011). Variability in the interpretation of chest radiographs for the diagnosis of pneumonia in children. *Journal of Hospital Medicine*, 7(4), pp.294–298. doi:<https://doi.org/10.1002/jhm.955>.
- OpenGenus IQ: Computing Expertise & Legacy. (2021). Inception V3 Model Architecture. [online] Available at: <https://iq.opengenus.org/inception-v3-model-architecture/>.

- Ortiz-Toro, C., García-Pedrero, A., Lillo-Saavedra, M. and Gonzalo-Martín, C. (2022). Automatic pneumonia detection in chest X-ray images using textural features. *Computers in Biology and Medicine*, p.105466. doi:<https://doi.org/10.1016/j.compbimed.2022.105466>.
- Pankratz, D.G., Choi, Y., Imtiaz, U., Fedorowicz, G., Anderson, J.R., Colby, T.V., Myers, J.L., Lynch, D.A., Brown, K.K., Flaherty, K.R., Steele, M.P., Groshong, S.D., Raghu, G., Barth, N.M., P. Sean Walsh, Huang, J., Kennedy, G.C. and Martínez, F.J. (2017). Usual Interstitial Pneumonia Can Be Detected in Transbronchial Biopsies Using Machine Learning. *Annals of the American Thoracic Society*, 14(11), pp.1646–1654. doi:<https://doi.org/10.1513/annalsats.201612-947oc>.
- pluralsight.(2023). Introduction to DenseNet with TensorFlow | Pluralsight. [online] Available at: <https://www.pluralsight.com/guides/introduction-to-densenet-with-tensorflow>.
- Puneet Gupta (2021). Pneumonia Detection Using Convolutional Neural Networks. January 2021, 7(01), pp.77–80. doi:<https://doi.org/10.46501/ijmtst070117>.
- Racic, L., Popovic, T., cakic, S. and Sandi, S. (2021). Pneumonia Detection Using Deep Learning Based on Convolutional Neural Network. 2021 25th International Conference on Information Technology (IT). doi:<https://doi.org/10.1109/it51528.2021.9390137>.
- Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Ball, R., Langlotz, C., Shpanskaya, K., Lungren, M. and Ng, A. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning CheXNet. [online] Available at: <https://arxiv.org/pdf/1711.05225.pdf>.
- Rakhlin, A., Shvets, A., Iglovikov, V. and Kalinin, A. (2018). Deep Convolutional Neural Networks for Breast Cancer Histology Image Analysis. [online] Available at: <https://arxiv.org/pdf/1802.00752.pdf> [Accessed 17 Jan. 2020].
- Sharma, S. and Guleria, K. (2023). A Deep Learning based model for the Detection of Pneumonia from Chest X-Ray Images using VGG-16 and Neural Networks. *Procedia Computer Science*, 218, pp.357–366. doi:<https://doi.org/10.1016/j.procs.2023.01.018>.
- Shorten, C. and Khoshgoftaar, T.M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, [online] 6(1). doi:<https://doi.org/10.1186/s40537-019-0197-0>.
- Sunil L. Bangare, Hrushikesh S. Rajankar, Pavan S. Patil, Karan V. Nakum and Gopal S. Paraskar (2022). Pneumonia Detection and Classification using CNN and VGG16. *International Journal of Advanced Research in Science, Communication and Technology*, pp.771–779. doi:<https://doi.org/10.48175/ijarsct-3851>.
- Tio, A.E. (2019). Face shape classification using Inception v3. arXiv (Cornell University).[online] Available at: <https://arxiv.org/ftp/arxiv/papers/1911/1911.07916.pdf>
- Tsai, M.-J. and Tao, Y.-H. (2019). Machine Learning Based Common Radiologist-Level Pneumonia Detection on Chest X-rays. doi:<https://doi.org/10.1109/icspcs47537.2019.9008684>.
- Varshni, D., Thakral, K., Agarwal, L., Nijhawan, R. and Mittal, A. (2019). Pneumonia Detection Using CNN based Feature Extraction. [online] IEEE Xplore. doi:<https://doi.org/10.1109/ICECCT.2019.8869364>.
- Vellaichamy, A.S., Swaminathan, A., Varun, C. and S, K. (2021). MULTIPLE PLANT LEAF DISEASE CLASSIFICATION USING DENSENET-121 ARCHITECTURE. *INTERNATIONAL*

JOURNAL OF ELECTRICAL ENGINEERING AND TECHNOLOGY, 12(5).

doi:<https://doi.org/10.34218/ijeet.12.5.2021.005>.

World Health Organization (2021). Pneumonia. [online] Who.int. Available at:

<https://www.who.int/news-room/fact-sheets/detail/pneumonia>.

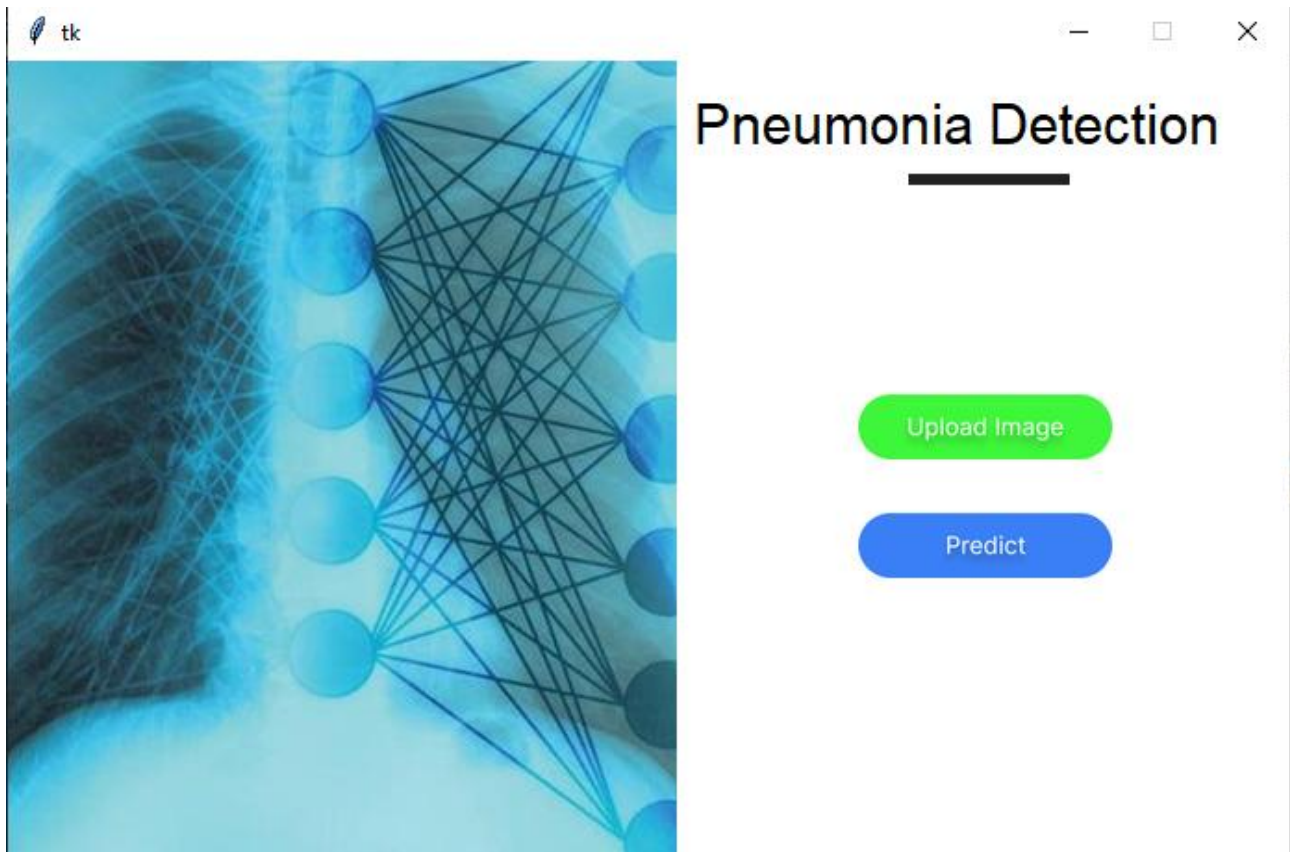
Wu, H., Xie, P., Zhang, H., Li, D. and Cheng, M. (2020). Predict pneumonia with chest X-ray images based on convolutional deep neural learning networks. *Journal of Intelligent & Fuzzy Systems*, pp.1–15. doi:<https://doi.org/10.3233/jifs-191438>.

Yao, S., Chen, Y., Tian, X. and Jiang, R. (2021). Pneumonia Detection Using an Improved Algorithm Based on Faster R-CNN. *Computational and Mathematical Methods in Medicine*, [online] 2021, p.8854892. doi:<https://doi.org/10.1155/2021/8854892>.

Appendices

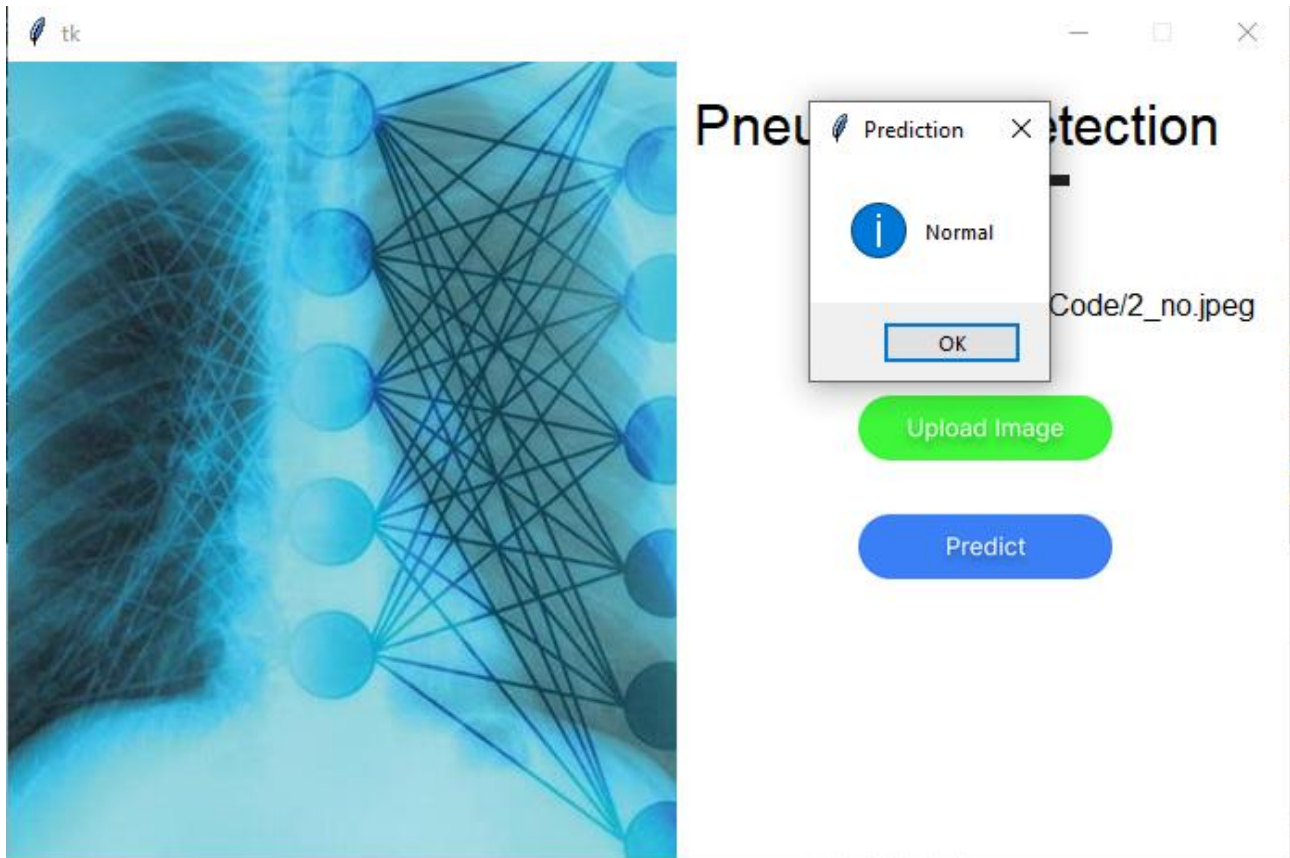
Appendix A

Home UI of the desktop application



Appendix B

The GUI for normal classification



Appendix C

The code :

Inception:

```
from google.colab import drive
drive.mount('/content/drive')

import os
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import InceptionV3
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D
from tensorflow.keras.models import Model
from sklearn.metrics import classification_report, confusion_matrix

train_data_dir = "/content/drive/My Drive/Aksa/Dataset/chest_xray/train"
test_data_dir = "/content/drive/My Drive/Aksa/Dataset/chest_xray/test"

train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

val_datagen = ImageDataGenerator(rescale=1./255)
```

```
batch_size = 32
```

```
train_generator = train_datagen.flow_from_directory(  
    train_data_dir,  
    target_size=(299, 299),  
    batch_size=batch_size,  
    class_mode='binary',  
    shuffle=True  
)
```

```
validation_generator = val_datagen.flow_from_directory(  
    test_data_dir,  
    target_size=(299, 299),  
    batch_size=batch_size,  
    class_mode='binary',  
    shuffle=False  
)
```

```
base_model = InceptionV3(weights='imagenet', include_top=False, input_shape=(299, 299, 3))
```

```
x = base_model.output  
x = GlobalAveragePooling2D()(x)  
x = Dense(1024, activation='relu')(x)  
predictions = Dense(1, activation='sigmoid')(x)
```

```
model = Model(inputs=base_model.input, outputs=predictions)
```

```
for layer in base_model.layers:
```

```
    layer.trainable = False
```

```
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

```
history = model.fit(
```

```
    train_generator,
```

```
    steps_per_epoch=len(train_generator),
```

```
    epochs=5,
```

```
    validation_data=validation_generator,
```

```
    validation_steps=len(validation_generator)
```

```
)
```

```
model.save('/content/drive/My Drive/Aksa/incep.h5')
```

```
plt.plot(history.history['loss'], label='Train Loss')
```

```
plt.plot(history.history['val_loss'], label='Val Loss')
```

```
plt.xlabel('Epochs')
```

```
plt.ylabel('Loss')
```

```
plt.legend()
```

```
plt.show()
```

```
plt.plot(history.history['accuracy'], label='Train Accuracy')
```

```
plt.plot(history.history['val_accuracy'], label='Val Accuracy')
```

```
plt.xlabel('Epochs')
```

```
plt.ylabel('Accuracy')
```

```
plt.legend()
```

```
plt.show()
```

```
y_true = validation_generator.classes  
y_pred_prob = model.predict(validation_generator)  
y_pred = y_pred_prob > 0.5
```

```
report = classification_report(y_true, y_pred, target_names=['NORMAL', 'PNEUMONIA'])  
print(report)
```

```
confusion_mat = confusion_matrix(y_true, y_pred)  
import seaborn as sns
```

```
sns.heatmap(confusion_mat, annot=True, fmt='d', cmap='Blues',  
            xticklabels=['NORMAL', 'PNEUMONIA'],  
            yticklabels=['NORMAL', 'PNEUMONIA'])
```

```
plt.title('Confusion Matrix')  
plt.xlabel('Predicted')  
plt.ylabel('Actual')  
plt.show()
```

ResNet:

```
from google.colab import drive  
drive.mount('/content/drive')
```

```
import os
```

```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import ResNet50
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D
from tensorflow.keras.models import Model
from sklearn.metrics import classification_report, confusion_matrix

train_data_dir = "/content/drive/My Drive/Aksa/Dataset/chest_xray/train"
test_data_dir = "/content/drive/My Drive/Aksa/Dataset/chest_xray/test"

train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

val_datagen = ImageDataGenerator(rescale=1./255)

batch_size = 32

train_generator = train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(224, 224),
    batch_size=batch_size,
    class_mode='binary',
```

```
    shuffle=True  
)
```

```
validation_generator = val_datagen.flow_from_directory(  
    test_data_dir,  
    target_size=(224, 224),  
    batch_size=batch_size,  
    class_mode='binary',  
    shuffle=False  
)
```

```
base_model = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))
```

```
x = base_model.output  
x = GlobalAveragePooling2D()(x)  
x = Dense(1024, activation='relu')(x)  
predictions = Dense(1, activation='sigmoid')(x)
```

```
model = Model(inputs=base_model.input, outputs=predictions)
```

```
for layer in base_model.layers:  
    layer.trainable = False
```

```
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

```
history = model.fit(
    train_generator,
    steps_per_epoch=len(train_generator),
    epochs=5,
    validation_data=validation_generator,
    validation_steps=len(validation_generator)
)

model.save('/content/drive/My Drive/Aksa/resnet.h5')

plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Val Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()

plt.plot(history.history['accuracy'], label='Train Accuracy')
plt.plot(history.history['val_accuracy'], label='Val Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()
plt.show()

y_true = validation_generator.classes
y_pred_prob = model.predict(validation_generator)
y_pred = y_pred_prob > 0.5
```

```
report = classification_report(y_true, y_pred, target_names=['NORMAL', 'PNEUMONIA'])  
print(report)
```

```
confusion_mat = confusion_matrix(y_true, y_pred)  
import seaborn as sns
```

```
sns.heatmap(confusion_mat, annot=True, fmt='d', cmap='Blues',  
            xticklabels=['NORMAL', 'PNEUMONIA'],  
            yticklabels=['NORMAL', 'PNEUMONIA'])
```

```
plt.title('Confusion Matrix')  
plt.xlabel('Predicted')  
plt.ylabel('Actual')  
plt.show()
```

Densenet:

```
from google.colab import drive  
drive.mount('/content/drive')
```

```
import os  
import numpy as np  
import matplotlib.pyplot as plt  
from tensorflow.keras.preprocessing.image import ImageDataGenerator  
from tensorflow.keras.applications import DenseNet121  
from tensorflow.keras.layers import Dense, GlobalAveragePooling2D  
from tensorflow.keras.models import Model  
from sklearn.metrics import classification_report, confusion_matrix
```



```
train_data_dir = "/content/drive/My Drive/Aksa/Dataset/chest_xray/train"
```

```
test_data_dir = "/content/drive/My Drive/Aksa/Dataset/chest_xray/test"
```

```
train_datagen = ImageDataGenerator(  
    rescale=1./255,  
    shear_range=0.2,  
    zoom_range=0.2,  
    horizontal_flip=True  
)
```

```
val_datagen = ImageDataGenerator(rescale=1./255)
```

```
batch_size = 32
```

```
train_generator = train_datagen.flow_from_directory(  
    train_data_dir,  
    target_size=(224, 224),  
    batch_size=batch_size,  
    class_mode='binary',  
    shuffle=True  
)
```

```
validation_generator = val_datagen.flow_from_directory(  
    test_data_dir,  
    target_size=(224, 224),  
    batch_size=batch_size,
```

```
class_mode='binary',
shuffle=False
)

base_model = DenseNet121(weights='imagenet', include_top=False)

x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(1024, activation='relu')(x)
predictions = Dense(1, activation='sigmoid')(x)

model = Model(inputs=base_model.input, outputs=predictions)

for layer in base_model.layers:
    layer.trainable = False

model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

history = model.fit(
    train_generator,
    steps_per_epoch=len(train_generator),
    epochs=5,
    validation_data=validation_generator,
    validation_steps=len(validation_generator)
)
```

```
model.save('/content/drive/My Drive/Aksa/dense.h5')
```

```
plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Val Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()
```

```
plt.plot(history.history['accuracy'], label='Train Accuracy')
plt.plot(history.history['val_accuracy'], label='Val Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()
plt.show()
```

```
y_true = validation_generator.classes
```

```
y_true = validation_generator.classes
y_pred_prob = model.predict(validation_generator)
y_pred = y_pred_prob > 0.5
```

```
report = classification_report(y_true, y_pred, target_names=['NORMAL', 'PNEUMONIA'])
print(report)
```

```
import seaborn as sns
```

```
confusion_mat = confusion_matrix(y_true, y_pred)

sns.heatmap(confusion_mat, annot=True, fmt='d', cmap='Blues',
            xticklabels=['NORMAL', 'PNEUMONIA'],
            yticklabels=['NORMAL', 'PNEUMONIA'])

plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

Ensemble:

import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

densenet_model = load_model('dense.h5', compile=False)
densenet_model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

resnet_model = load_model('resnet.h5', compile=False)
resnet_model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

inception_model = load_model('incep.h5', compile=False)

inception_model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

densenet_input_shape = (224, 224, 3)
```

```
resnet_input_shape = (224, 224, 3)
```

```
inception_input_shape = (299, 299, 3)
```

```
def preprocess_input(model, img_path, input_shape):
```

```
    img = image.load_img(img_path, target_size=(input_shape[0], input_shape[1]))
```

```
    img = image.img_to_array(img)
```

```
    img = np.expand_dims(img, axis=0)
```

```
    img = img / 255.0
```

```
    return img
```

```
densenet_img = preprocess_input(densenet_model, '2_pn.jpeg', densenet_input_shape)
```

```
resnet_img = preprocess_input(resnet_model, '2_pn.jpeg', resnet_input_shape)
```

```
inception_img = preprocess_input(inception_model, '2_pn.jpeg', inception_input_shape)
```

```
densenet_pred = densenet_model.predict(densenet_img)
```

```
resnet_pred = resnet_model.predict(resnet_img)
```

```
inception_pred = inception_model.predict(inception_img)
```

```
print(densenet_pred)
```

```
print(resnet_pred)
```

```
print(inception_pred)
```

```
ensemble_pred = (densenet_pred + resnet_pred + inception_pred) / 3.0
```

```
print(f'Ensemble Prediction: {ensemble_pred}')
```

```
a=ensemble_pred[0][0]
```

```
if (a>0.5):
```

```
    print("Pneumonia")
```

```
else:
```

```
    print("Normal")
```

GUI:

```
from tkinter import Tk, Canvas, Button, Entry, PhotoImage, filedialog, messagebox
```

```
from pathlib import Path
```

```
from mo_predict import prd
```

```
selected_image_path = ""
```

```
def select_image():
```

```
    global selected_image_path
```

```
    selected_image_path = filedialog.askopenfilename(filetypes=[("JPEG files", "*.jpg *.jpeg")])
```

```
    if selected_image_path:
```

```
        entry_path.delete(0, "end")
```

```
        entry_path.insert(0, selected_image_path)
```

```
        messagebox.showinfo("Success", "Image selected successfully!")
```

```
def check_image():
```

```
    global selected_image_path
```

```
    if not selected_image_path:
```

```
        messagebox.showwarning("Error", "Please select an image first!")
```

```
    else:
```

```
        res=prd(selected_image_path)
```

```
        if res=="pn":
```

```
            messagebox.showinfo("Prediction", "Pneumonia Detected")
```

else:

 messagebox.showinfo("Prediction", "Normal")

window = Tk()

window.geometry("692x430")

window.configure(bg = "#FFFFFF")

canvas = Canvas(

 window,

 bg = "#FFFFFF",

 height = 430,

 width = 692,

 bd = 0,

 highlightthickness = 0,

 relief = "ridge"

)

canvas.place(x = 0, y = 0)

entry_path = Entry(

 window,

 font=("Arial", 12),

 width=50,

 readonlybackground="#FFFFFF",

 relief="flat"

)

entry_path.place(x=440, y=120)

```
button_image_1 = PhotoImage(
    file="assets/frame0/button_1.png")
button_1 = Button(
    image=button_image_1,
    borderwidth=0,
    highlightthickness=0,
    command=lambda: check_image(),
    relief="flat"
)
button_1.place(
    x=459.0,
    y=242.99999999999994,
    width=137.0,
    height=36.0
)

button_image_2 = PhotoImage(
    file="assets/frame0/button_2.png")
button_2 = Button(
    image=button_image_2,
    borderwidth=0,
    highlightthickness=0,
    command=lambda: select_image(),
    relief="flat"
)
button_2.place(
    x=459.0,
    y=178.99999999999994,
    width=137.0,
```



```
        height=36.0
    )

    canvas.create_text(
        370.0,
        16.999999999999943,
        anchor="nw",
        text="Pneumonia Detection",
        fill="#000000",
        font=("Inter Bold", 30 * -1)
    )

    canvas.create_rectangle(
        486.0,
        60.999999999999994,
        573.0,
        66.999999999999994,
        fill="#232222",
        outline="")

    image_image_1 = PhotoImage(
        file="assets/frame0/image_1.png")
    image_1 = canvas.create_image(
        180.0,
        215.000000000000006,
        image=image_image_1
    )
    window.resizable(False, False)
    window.mainloop()
```

Appendix D

Certificate Of Ethics Approval

Enhancing Pneumonia Detection: Using Ensemble Learning in Deep Learning-Based Models.

P164395



Certificate of Ethical Approval

Applicant:	Aksa George
Project Title:	Enhancing Pneumonia Detection: Using Ensemble Learning in Deep Learning-Based Models.

This is to certify that the above named applicant has completed the Coventry University Ethical Approval process and their project has been confirmed and approved as Low Risk

Date of approval:	13 Oct 2023
Project Reference Number:	P164395