

Article

Not peer-reviewed version

Bias-Resilient Ensemble of Transfer Learning Models for Automated COVID-19 Detection from Chest Radiographs

[Shahzad Ahmad Qureshi](#)*, [Muhammad Abdul Basit](#), Qasim Ijaz, [Syed Taimoor Hussain Shah](#),
Syed Mesam Tamar Kazmi, [Inam ul Haq](#), Ghulam Mustafa, [Syed Adil Hussain Shah](#)

Posted Date: 22 August 2025

doi: 10.20944/preprints202508.1553.v1

Keywords: COVID-19; chest X-rays; transfer learning; deep learning; ensemble; weighted optimization



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Bias-Resilient Ensemble of Transfer Learning Models for Automated COVID-19 Detection from Chest Radiographs

Shahzad Ahmad Qureshi ^{1,2,*}, Muhammad Abdul Basit ³, Qasim Ijaz ¹,
Syed Taimoor Hussain Shah ⁴, Syed Mesam Tamar Kazmi ⁵, Inam ul Haq ⁵, Ghulam Mustafa ⁵
and Syed Adil Hussain Shah ^{6,7}

- ¹ Department of Computer and Information Sciences, Pakistan Institute of Engineering and Applied Sciences (PIEAS), Islamabad 45650, Pakistan
 - ² Centre for Mathematical Sciences, PIEAS, Islamabad 45650, Pakistan
 - ³ Department of Mechanical Engineering, PIEAS, Islamabad 45650, Pakistan
 - ⁴ Department of Mechanical and Aerospace Engineering, Politecnico di Torino, 10129 Turin, Italy
 - ⁵ Department of Physics and Applied Mathematics, PIEAS, Islamabad 45650, Pakistan
 - ⁶ Department of Research and Development (R&D), GPI SpA, 38123 Trento, Italy
 - ⁷ PolitoBIOMed Lab, Department of Mechanical and Aerospace Engineering, Politecnico di Torino, 10129 Turin, Italy
- * Correspondence: drsaqureshi@pieas.edu.pk

Abstract

The viral disease COVID-19, declared a pandemic by the World Health Organization (WHO), primarily affects the respiratory system and can be fatal. Although the Reverse Transcriptase Polymerase Chain Reaction (RT-PCR) test remains the gold standard for COVID-19 diagnosis, its time-intensive nature limits its effectiveness in urgent situations. To address this, we propose an ensemble of five state-of-the-art transfer learning (TL) models designed to mitigate biases and enhance the classification of COVID-19 from chest radiographs. A weighted optimization strategy combines the models, giving more weight to those with superior performance, ensuring more accurate and robust predictions. Evaluated on a publicly available dataset, the SqueezeNet model achieved the highest accuracy of 94.01% for three-class classification (Normal, COVID-19, Lung Opacity), while the ensemble approach achieved 92.57% accuracy and an F1-score of 92.36%, demonstrating resilience to transfer learning biases. This framework offers reliable diagnostic support, streamlining radiology workflows and enhancing decision-making in high-demand clinical environments. Additionally, it serves as a valuable tool for advancing medical artificial intelligence expertise among graduates.

Keywords: COVID-19; chest X-rays; transfer learning; deep learning; ensemble; weighted optimization

1. Introduction

This project aims at an accurate and efficient detection of COVID-19 (C19) carried out for the design and development of advanced deep learning (DL) and transfer learning (TL) techniques. These techniques use large datasets of medical images, such as CT scans and chest X-rays, to identify anomalies and patterns associated with the virus. By training neural networks to recognize these specific features, the system can assist in rapid screening and diagnosis. The goal is to provide a reliable, non-invasive, and cost-effective diagnostic tool that can serve as an alternative or supplement to the traditional Reverse Transcription Polymerase Chain Reaction (RT-PCR) test, which, although widely used, often requires more time and specialized laboratory equipment, has a

reported sensitivity (S_n) of approximately 59%, and can be prone to false negatives [1]. The implementation of AI-driven diagnostic methods could significantly enhance early detection, especially in resource-constrained settings or during mass outbreaks.

Since the outbreak of C19 in December 2019, the pandemic has had a devastating global impact, resulting in more than 7.1 million reported deaths and over 776 million confirmed cases to date [2]. Building on a foundational grasp of machine learning (ML) principles, algorithms, and their healthcare applications, this work incorporated an extensive review of literature on C19 and related respiratory illnesses such as viral pneumonia (VP), focusing on their clinical and radiological distinctions. Understanding disease characteristics and progression patterns was vital for addressing diagnostic complexities. Subsequently, DL approaches were explored for their potential in analyzing medical images to detect C19 and similar conditions. This integrated study provided a coherent framework for using advanced computational models to enhance diagnostic accuracy and support informed clinical decision-making.

This study presents advanced neural network methods for rapid and accurate C19 detection from chest radiographs, supporting clinicians and aiding medical training. It addresses diagnostic challenges across age groups, especially differentiating scans of young and elderly patients. C19, caused by SARS-CoV-2, is a highly contagious respiratory disease marked by lung inflammation and symptoms like breathing difficulty, fever, and dry cough [3]. Since its outbreak, it has infected 16–28 million people globally, with older adults at higher risk due to weaker immunity and comorbidities, while children often show mild or no symptoms, complicating diagnosis [2]. This approach aims to improve early detection, essential for effective disease control.

C19 symptoms vary widely, with fever reported in about 81% of patients, cough in 58%, and dyspnea and sputum production in roughly 25% of cases. Patients also commonly experience headaches, breathing difficulties, and fatigue, which may worsen as the infection advances. Accurate diagnosis depends on laboratory and imaging tests: RT-PCR is the gold standard for detecting viral genetic material, while radiological tools like CT scans and chest X-rays identify lung abnormalities such as ground glass opacities. Serological tests help detect prior exposure and support epidemiological studies.[3]. Treatment options vary according to disease severity, including corticosteroids to reduce inflammation, anticoagulants to prevent blood clots, supplemental oxygen, and mechanical ventilation for severe respiratory failure. Additionally, rehabilitation therapies assist recovery from severe illness and long C19 symptoms, aiming to restore lung function and improve quality of life [4].

While C19 and pneumonia share respiratory symptoms, they differ in cause, lung involvement, and radiographic features. C19, caused by SARS-CoV-2, usually affects both lungs simultaneously, especially the lower lobes, whereas pneumonia, caused by bacteria, viruses, or fungi, often presents as a localized infection in one lung with a more diffuse or patchy pattern. On chest X-rays, C19 consolidations appear denser and irregular, primarily in lower zones, while pneumonia consolidations are less dense and better defined. Pleural effusion is generally minimal or absent in C19 but more common and pronounced in bacterial pneumonia, signaling greater inflammation or complications. These differences are vital for accurate diagnosis and treatment planning [5].

C19 can cause multiple complications, especially in moderate to severe cases or those with pre-existing conditions. Common issues include pneumonia, characterized by lung inflammation and fluid buildup often affecting both lungs, lung opacity (LO) visible on imaging, and in severe cases, Acute Respiratory Distress Syndrome (ARDS), which necessitates intensive care. Bronchitis and long-term pulmonary fibrosis may also develop, impairing lung function and breathing. Early diagnosis and management are crucial to reduce such complications. Treatment faces challenges due to limited antiviral efficacy and the continuous emergence of SARS-CoV-2 variants that can evade immune responses, complicating vaccination and therapy efforts. Vaccine hesitancy and misinformation further hinder control measures. Diagnostic hurdles include the lengthy RT-PCR testing process, limited ability to detect new variants, and radiological similarities between C19 and other respiratory illnesses like pneumonia and bronchitis, making differential diagnosis difficult.

In this study, we employed large deep learning architectures with pretrained weights in TL mode. Based on prior research, we selected five algorithms as potential backbones. The results of each architecture are post-processed using an ensemble weight-optimized distribution algorithm, which mitigates model bias and provides more reliable results for C19 patients, making it clinically preferable to a single DL strategy. Outlined below are the principal contributions of this article.

1. The article proposes the design and implementation of DL-based TL methods tailored for the robust detection of C19 from chest radiographs and CT scans using a weight-optimized distribution algorithm. The metrics, from each of the large DL architectures, are normalized by dividing each metric by the sum of the corresponding metrics from all models. This normalization ensures that each metric's influence is proportional to the individual performance of each of the five models.
2. It uses an extensive dataset with variations to make it more challenging. The good classification ability of Class A could negatively impact the performance of Classes B and C due to feature dominance. We have used the first three classes (C19, LO, N) to make the task challenging. The last class achieves outstanding results, specifically, due to its discriminative characteristics, which can otherwise lead to negatively biased performance.
3. It explicitly tackles the variation in imaging features across five different DL algorithms, improving differentiation between C19 manifestations in young versus elderly patients.
4. The study addresses limitations of the traditional RT-PCR test, such as lower Sn (~59%), longer turnaround times, and the need for specialized laboratories, by offering a non-invasive, cost-effective AI-based diagnostic tool suitable for resource-limited settings.
5. Support for Clinical Decision-making and Medical Education: Beyond assisting radiologists with a reliable second opinion in image interpretation, the proposed framework presents a valuable educational tool for medical graduates.

This article is organized into four sections. Section 1 provides an introduction, outlining the background and significance of the study on C19. The materials and methods used in this work are detailed in Section 2, whereas the Section 3 presents the results and a comprehensive discussion, evaluating the performance of the proposed model in comparison to existing methods. Finally, Section 4 concludes the article by summarizing the key findings and discussing the potential implications of the research.

1.1. Related Work

C19, caused by the SARS-CoV-2 virus, is a rapidly spreading infectious disease that has led to a global health crisis and significant mortality worldwide [6]. Early symptoms commonly include fever, respiratory distress, pneumonia, and reduced white blood cell count. The RT-PCR technique is the primary diagnostic method, as it detects viral RNA; however, a negative result does not fully rule out infection. Consequently, medical imaging modalities such as chest X-rays are often employed as supplementary diagnostic tools [7]. The integration of AI into healthcare has expanded considerably with the increasing availability of large-scale datasets. Using extensive collections of chest X-ray and CT images, AI can facilitate disease detection, predict patient outcomes, and recommend optimal treatment strategies, thereby enhancing diagnostic accuracy and efficiency [8]. In this context, researchers have extensively explored deep neural networks and ensemble learning techniques for their potential in detecting C19 from chest radiographs.

Jouibari et al. [9] utilized a private dataset from a local hospital containing both emergency and routine cases. They applied TL with 11 well-known neural networks, with ResNet50 achieving an accuracy of 95.00%. To improve classification between cases, ensemble-learning methods such as majority voting and error-correcting output codes were employed. Similarly, Srinivas et al. [10]

worked with a dataset of 243 chest X-rays, including 191 COVID-positive and 122 Normal (N) cases. Their hybrid V3-VGG architecture comprised four blocks: the first used VGG16, the second and third employed InceptionV3, and the fourth incorporated average pooling, dropout, a fully connected layer, and Softmax. This model reached 98.00% accuracy.

Kolhar et al. [8] evaluated the effectiveness of DL models VGG16 and ResNet50 in detecting tuberculosis, C19, and healthy lungs from chest X-rays. ResNet50 demonstrated precision and recall rates close to 99.00%, while VGG16 performed well in tuberculosis detection. Kumar et al. [11] used CNNs to automatically extract features, removing the need for manual feature selection. Their CXNet-based model classified images as N, C19, or pneumonia with an accuracy of 98.00%.

Fernandaz-Miranda et al. [12] trained the VGG16 CNN model on three datasets from different institutions. Although hyperparameters remained consistent, the model's performance dropped by up to 8.00% when images from a different manufacturer were included. Variations in device response and image processing led to performance reductions of 18.90% and 9.80% in separate instances. Brunese et al. [13] proposed a three-phase approach: detecting pneumonia presence, classifying C19 and pneumonia using VGG16, and localizing affected regions with GRAD-CAM. Using a dataset of 6,523 images, they achieved 93.00% accuracy.

Nayak et al. [14] conducted a comprehensive comparison of DL models such as AlexNet, VGG16, and ResNet-34, finding ResNet-34 to be the most effective with 98.33% accuracy, making it a strong candidate for early C19 detection. Correspondingly, Zhang et al. [15] applied DL to 1,531 chest X-rays, achieving 96.00% accuracy for C19 cases and 70% for non-COVID cases. They also used Grad-CAM for lung region localization. Arias-Londono et al. [16] investigated the impact of data preprocessing on model performance, demonstrating that appropriate preprocessing of 79,500 X-ray images enhanced both accuracy and explainability.

Similarly, Vaid et al. [17] employed TL with CNNs on publicly available chest radiographs, reaching 96.30% accuracy. Their confusion matrix showed 32 true positives for C19, confirming the model's clinical reliability. Kassania et al. [18] compared feature extraction frameworks and identified that DenseNet121 combined with a Bagging Tree classifier yielded the highest accuracy of 99.00% for C19 detection.

Hussain et al. [19] developed CoroDet, achieving 99.10% accuracy in detecting C19, N, and pneumonia cases. In three-class case (C19, N, pneumonia), the model reached 94.30% accuracy, while in four-class classification (C19, N, VP, bacterial pneumonia (BP)) it achieved 92.10%. Similarly, Ismael et al. [20] employed ResNet50 for C19 classification, attaining 94.70% accuracy with fine-tuned models.

Xu et al. [21] proposed a DL approach for differentiating C19 from influenza-A VP and healthy cases using pulmonary CT images. Their method first localized infection regions with a 3D DL model, then classified C19, IAVP, and non-infectious cases using a location-attention classification model. The Noisy-or Bayesian function was finally applied to determine the infection type for each CT case.

Ali Abbasian et al. [22] optimized CNNs via TL, using ten architectures, AlexNet, VGG16, VGG-19, SqueezeNet, GoogleNet, MobileNet-V2, ResNet18, ResNet50, ResNet-101, and Xception, to distinguish C19 infections from non-C19 cases. Likewise, Li et al. [23] designed the C19 detection neural network (COVNet) to classify C19, community-acquired pneumonia (CAP), and non-pneumonia cases. Lung regions were extracted using the U-Net [24] segmentation method, and preprocessed images were passed to COVNet for predictions. Grad-CAM was applied to visualize key decision-making regions, improving model interpretability.

Hu et al. [25] utilized a ShuffleNet V2-based CNN for C19 detection from chest CT images. The model extracted features from 1,042 CT scans, comprising 397 healthy, 521 C19, 76 bacterial pneumonia, and 48 SARS cases, and achieved 91.21% accuracy on an independent test set. Similarly, Wang et al. [26] introduced COVID-Net, a deep CNN for C19 detection from chest X-ray (CXR) images, trained on the COVIDx dataset. The model initially classified non-infectious, non-C19, and C19 viral infection images, with GSInquire used to localize regions of interest. Generative synthesis was then applied to optimize both micro- and macro-architecture designs for COVID-Net.

Charmaine et al. [27] compared multiple CNN models for classifying CT scans into C19, influenza VP, or non-infectious categories. The pipeline involved preprocessing CT images to identify lung regions, applying a 3D CNN for segmentation, classifying the infection type, and generating an analysis report for each sample using the Noisy-or Bayesian function.

Rohit et al. [28] proposed a weighted consensus model to minimize false positives and false negatives in C19 detection. Within this framework, ML algorithms including Linear Regression, k-Nearest Neighbor, Random Forest, Support Vector Machine, and Decision Tree were applied. Each model predicted different classes, and normalized accuracy was computed for each class. Similarly, Sinra et al. [29] employed an ensemble ML approach to classify chest X-ray images into N, C19, and LO categories. They used the random forest classifier with 5-fold CV, and applied thresholding for image segmentation.

Ahmad et al. [30] designed a custom convolutional neural network (CNN) consisting of eight weighted layers, incorporating dropout and batch normalization (BN), and trained using stochastic gradient descent for 30 epochs. The model classified chest X-ray samples into C19, N, and pneumonia categories, achieving a precision score of 98.19%. Chen et al. [31] applied TL to classify X-ray images into C19, VP, LO, and N cases. They evaluated Efficient Neural Networks, multi-scale Vision Transformers, Efficient Vision Transformers, and standard Vision Transformers on 3616 C19, 6012 LO, 10192 N, and 1345 VP samples. The Vision Transformer achieved the highest accuracy, 98.58% (binary classification), 99.57% (three-class), and 95.79% (four-class).

Alamin et al. [32] explored fine-tuning across various architectures using approximately 2,000 X-ray images. Xception, InceptionResNetV2, ResNet50, ResNet50V2, EfficientNetB0, and EfficientNetB4 achieved accuracies of 99.55%, 97.32%, 99.11%, 99.55%, 99.11%, and 100%, respectively, with EfficientNetB4 yielding the highest accuracy and an F1-score of 99.14%. Kirti et al. [33] fine-tuned four deep TL models, ResNet50, DenseNet, VGG16, and VGG19, using an augmented dataset. VGG-19 delivered the best performance across both datasets.

Sunil et al. [34] introduced CovidMediscanX, a framework combining custom CNN models with pre-trained and hybrid TL architectures for C19 detection from chest X-rays. The custom CNN achieved 94.32% accuracy, though its lower recall in labeling N cases indicated the need for further refinement. Yadlapali et al. [35] combined fuzzy logic with DL to differentiate C19 pneumonia from interstitial pneumonia, employing a ResNet18 four-class classifier for C19, VP, N, and bacterial pneumonia. This approach achieved 97.00% accuracy, 96.00% precision, and 98.00% recall, outperforming other methods. Similarly, Dokumaci et al. [36] classified X-ray images into C19, LO, and VP using a four-way classifier. Among evaluated CNN models, ConvNext achieved the highest performance, with 98.10% accuracy and 97.80% precision, demonstrating strong potential for early C19 diagnosis.

Our literature review was key to shaping this research. It helped us see what worked well in past studies and where the gaps were. This guided our choice of preprocessing methods, like bilateral filtering and discrete wavelet transform, to improve image quality. We also identified strong DL models, such as VGG16 and ResNet50, for our classification tasks.

2. Materials and Methods

The proposed methodology for C19 detection from chest radiographs integrates preprocessing, data augmentation, and TL techniques within a 5-fold CV framework as illustrated in Figure 1. The dataset comprises chest X-ray images taken from publicly available C19 repository, containing both COVID-positive and non-COVID cases. The preprocessing stage involves converting RGB images to grayscale (RGB2GL), resizing to 512 × 512 pixels, applying wavelet transform (WT) to reduce dimensions to 256 × 256 pixels, and performing bilateral filtering (BF) for noise reduction, smoothing the images while preserving the edges. Augmentation techniques, rotation, reflection, translation, and affine transformation, are employed to improve model generalization. Five TL architectures, AlexNet, ResNet18, ResNet50, SqueezeNet, and VGG16, are applied to the dataset, with an optimized weight distribution algorithm implemented to mitigate bias in the DL models and ensure robust

performance. Model evaluation is conducted using accuracy, F1-score, Sn, specificity (Sp), ROC and PR curves, precision, and negative predictive value (NPV).

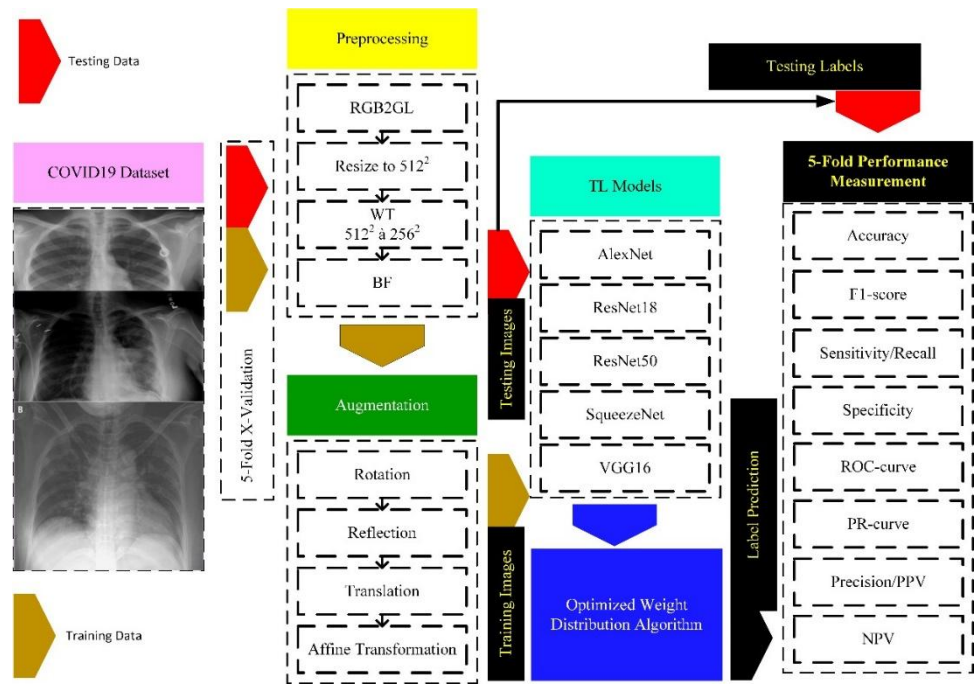


Figure 1. Methodological Framework for C19 detection using DL Models.

2.1. Dataset

A collaborative project between the University of Dhaka, Bangladesh, and Qatar University, Doha, Qatar, with contributions from researchers in Malaysia and Pakistan and guidance from medical experts, compiled a chest X-ray dataset covering C19, LO, and N cases [37,38]. The dataset contains 19820 images: 3616 C19, 6012 LO, and 10192 N cases. Table 1 summarizes the dataset composition, while Figure 2 illustrates the class imbalance. To increase the challenge of the dataset, we have excluded the highly distinguishing class of viral pneumonia (VP), having 1345 samples (Section 3.8).

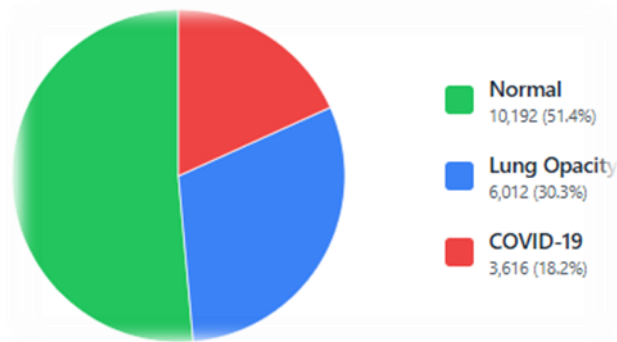


Figure 2. C19 dataset: instance distribution by class.

Table 1. Outline of the C19 Radiography Dataset containing 19,820 samples.

Feature	Dataset
Dataset name	C19 Radiography database
Year	2020
Number of subjects	3616 C19 patients

Availability	Publicly available
Site address	https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database

2.2. Preprocessing

Preprocessing prepares raw data for ML and DL by cleaning, normalizing, and transforming it. It's essential to ensure data quality, reduce noise, improve model accuracy, and speed up training. All input images are stored in PNG format. They undergo a structured preprocessing pipeline comprising RGB to grey-scale conversion, resizing, wavelet decomposition, and bilateral filtering before being used for model training and evaluation.

2.2.1. RGB to Grey-Scale Conversion

The original RGB images converted to grey-scale using a weighted sum of the R, G, and B channels. This reduces computational complexity by reducing the input from three color channels to one, while preserving key luminance information [39].

2.2.2. Resizing the Scans

To ensure uniformity in input dimensions, all grey-scale images resized to 512×512 pixels using interpolation techniques. For image resizing, bilinear interpolation was used. In this method, the position of each pixel in the resized image is mapped back to the original image, and the final pixel value is computed based on the weighted contribution of four surrounding pixels, represented as p , q , r , and s , as shown in Figure 3, given by (k,l) , $(k,l+1)$, $(k+1,l)$, and $(k+1,l+1)$ with γ coordinate as (s,t) . Bilinear interpolation is performed by computing the influence of p and q , and representing as α , as given by $f(k,l+t) = [f(k,l+1) - f(k,l)]t + f(k,l)$. Next, compute the influence of r and s , and represent it as β , given by $f(k+1,l+t) = [f(k+1,l+1) - f(k+1,l)]t + f(k+1,l)$. Finally, the effect of α and β is computed, and denoted as γ , given by $f(k+s,l+t) = (1-s)(1-t)f(k,l) - (k-s)t f(k,l+1) + s(1-t)f(k+1,l) + st f(k+1,l+1)$. This method uses horizontal and vertical interpolation, surpassing nearest neighbor in quality while being faster than bicubic interpolation [39].

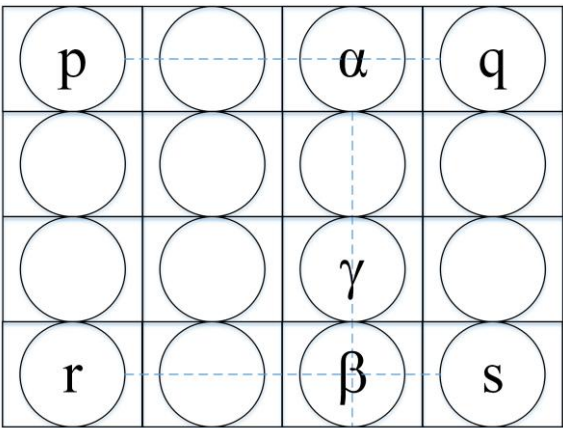


Figure 3. Resizing the chest X-ray images using bilinear interpolation where the intensity value at a new pixel location is estimated based on the weighted average based on the four nearest surrounding pixels.

2.2.3. Discrete Wavelet Transform

To improve contrast in chest X-ray images, we applied the Discrete Wavelet Transform (DWT) with a Level-2 decomposition using low-pass and high-pass filter banks [40]. This yielded sub-band images: LL, LH, HL, and HH as illustrated in Figure 4. We retained the approximation (LL) and

diagonal detail (HH) components, performed inverse DWT to return to Level-1, and merged them into a 256×256 DWT image. The image was then down-sampled to 224×224 for consistency, with LH and HL components excluded due to their minimal impact on contrast.

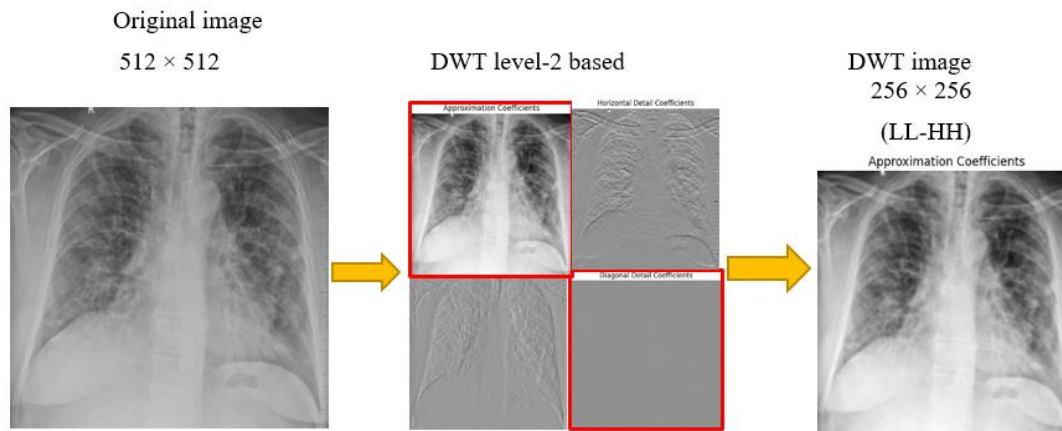


Figure 4. DWT decomposition of X-ray radiographs to obtain high contrast images.

2.2.4. Bilateral Filtering

We applied bilateral filtering to our dataset to smooth images while preserving edges. This technique operates through a non-linear combination of nearby pixel values and is characterized by its non-iterative, local, and computationally simple nature. Bilateral filtering combines pixel intensities based on both their spatial proximity (geometric closeness) and intensity similarity (photometric similarity), assigning greater weight to values that are both spatially and radiometrically closer. As a result, it effectively reduces noise without blurring important structural edges [41].

The bilateral filter is defined as:

$$I^{\text{filtered}}(x) = \frac{1}{W_p} \sum_{x_i \in \emptyset} I(x_i) f_r(|I(x_i) - I(x)|) g_s(|x_i - x|) \quad (1)$$

and normalized term W_p is defined as:

$$W_p = \sum_{x_i \in \emptyset} f_r(|I(x_i) - I(x)|) g_s(|x_i - x|), \quad (2)$$

where, I^{filtered} is the resulting filtered image, I is the image in original form, x represents the coordinates of the pixel currently undergoing filtering, $\emptyset$ is the window centred in x , so $x_i \in \emptyset$ is another pixel, f_r denotes the range kernel, which smooths based on intensity differences between pixels, g_s is the spatial kernel responsible for smoothing based on the spatial proximity of pixels (often modeled with a Gaussian function).

The Table 2 outlines the parameters used for bilateral filtering, specifically chosen to achieve optimal performance. The first parameter, diameter (D), determines the size of the pixel neighborhood considered during filtering. A larger diameter results in a stronger smoothing effect, with a selected value of 9 for this particular setup. The SigmaColor (σ_{color}) parameter controls how much color differences influence the filtering process. For gray level images, there is lacking color information and consists only of intensity (brightness) values, sigmaColor essentially controls how much intensity difference between neighboring pixels influences the filter. A higher value results in more smoothing, and in this case, a value of 75 was chosen to enhance the color Sn of the filter. Lastly, SigmaSpace (σ_{space}) specifies how far in space the filter considers when applying smoothing. A larger value indicates that the filter will take into account more distant pixels, which results in broader smoothing. Here, a value of 75 was selected, allowing the filter to include a larger spatial range in its smoothing effect. These parameters, when fine-tuned as described, are designed to

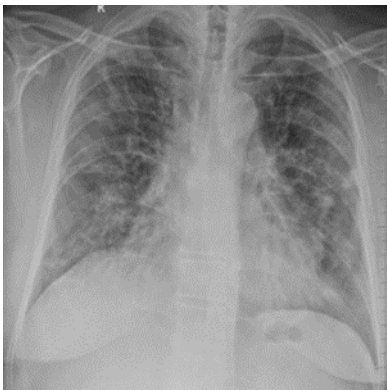
optimize both edge preservation and the overall quality of the filtering process. The result of applying the bilateral filter to the original image, shown in Figure 5(a), is illustrated in Figure 5(b).

Table 2. Parameterization of bilateral filtering optimized for optimal performance.

Parameter	Effect	Selected Value
D	Diameter of the pixel neighborhood. Larger values mean a stronger effect.	9
sigmaColor	Color differences affect filtering; a larger value means more smoothing.	75
sigmaSpace	How far in space does the filter look? Larger value considers more distant pixels.	75



(a) Original image



(b) Bilateral filtered image

Figure 5. Bilateral filtered image (a) is very smooth with preserved edges, reduced texture than (a) original image.

2.2.5. Augmentation

To address dataset imbalance, a variety of geometric transformations were applied, including rotation, translation, reflection, and affine adjustments as illustrated in Table 3. Images were rotated clockwise or anti-clockwise within a range of ± 10 degrees to introduce orientation variability while preserving key features. Translation shifts of up to $\pm 10\%$ of the image dimensions were applied both horizontally and vertically, simulating minor positional changes without distorting the content. Horizontal reflection (flipping across the X-axis) was randomly performed with a 50% probability ($p=0.5$), effectively increasing sample diversity for symmetrical objects. Additionally, random affine transformations were used, though limited to translation only, excluding shear, to maintain natural image geometry. These transformations enhanced dataset variety, reducing overfitting thereby improving generalization.

Table 3. Geometric transformations, including rotation, reflection, translation, and affine adjustments, applied to address dataset imbalance.

Type	Variation	Extreme Range / Value
Rotation	Clockwise / Anti-clockwise	± 10 degrees
Translation	Horizontal and Vertical shift	$\pm 10\%$ of image dimensions
Reflection	Horizontal (X-axis)	Random flip ($p=0.5$)
Affine	Random affine transformation	Translation only (no shear)

2.3. DL Overview

In this section, we provide a comprehensive overview of the fundamental concepts, and processes to understanding DL architectures. It involves multiple layers of interconnected nodes (neurons) that process input data hierarchically, allowing the model to learn increasingly abstract representations. Key components such as input layers, hidden layers, output layers, activation functions), loss functions, and optimization algorithms are discussed.

2.3.1. Minibatch Size

It refers to the number of incidents in the dataset after which the optimizer updates the parameters. It can be a minibatch, or even the whole training set can be used as a batch for parameter optimization. Often, a minibatch is recommended for better performance [42].

2.3.2. Dropout

This is a regularization technique used to prevent over-fitting of DL algorithms, in which some random percentage of the neurons from each layer do not pass their information to the next layer during each training step [42].

2.3.3. Activation Functions

Functions used in the neuron that operate on the weighted sum and produce the final output of the neuron are called activation functions [43]. ReLU (Rectified Linear Unit) is a popular non-linear activation function in neural networks. Its key advantage is sparse activation, only neurons with positive outputs remain active, while others are deactivated [44]. Mathematically, it is defined as

$$f(x) = \max(0, x) \quad (3)$$

The ReLU maps the output x greater than or equal to 0; if $x > 0$, it remains same otherwise 0, as illustrated in Figure 6.

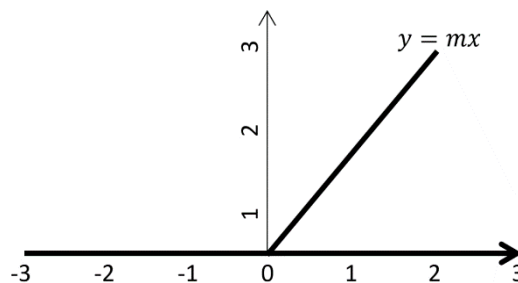


Figure 6. ReLU for input values $x < 0$, the function outputs 0 (a flat horizontal line along the x-axis), effectively suppressing negative values.

2.3.4. Softmax

The softmax function extends the sigmoid function for multiclass classification. While sigmoid outputs probabilities (0 to 1) for binary tasks, softmax computes probabilities for multiple classes, ensuring all outputs sum to 1 [45]. It is defined as.

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (4)$$

2.3.5. Loss Function

The loss function quantifies the difference between a model's predictions and the actual outputs [46]. For our multiclass classification task, we employed the cross-entropy (CE) loss function.

$$CE = - \sum_{c=1}^n y_c \log(p_c) \quad (5)$$

where n represents the total number of classes and y_c is the true label and p_c is the predicted probability of class c .

2.3.6. Convolution Layers

The layers, which apply a filter to the input to extract spatial information from input data for image and video recognition, are called convolutional layers. The filter is a small matrix, applied to the entire data for feature extraction [47].

$$h_j^{(n)} = \sum_{k=1}^k h_k^{(n-1)} * w_{kj}^{(n)} + b_j^{(n)} \quad (6)$$

where $h_j^{(n)}$ is the j^{th} feature map output in layer $h^{(n)}$, $h_k^{(n-1)}$ is the k^{th} channel in $h^{(n-1)}$, $w_{kj}^{(n)}$ is the k^{th} channel of the j^{th} filter in $h^{(n)}$ and $b_j^{(n)}$ is the corresponding bias term.

2.3.7. MaxPooling

The most common pooling method is MaxPooling, which extracts patches from input feature maps, retains the maximum value in each patch, and discards the rest [48]. Typically, a 2×2 filter with a stride of 2 is used, reducing the feature maps' spatial dimensions by half.

2.3.8. Fully Connected Layers

A fully connected (or dense) layer links every neuron to all neurons in the previous layer, each with its own weight. These layers enable maximum information flow [47]. For an input vector x , the layer performs a linear transformation, producing the output as

$$y = f(Wx + b), \quad (7)$$

where W is a weighted matrix, b is the bias term, and y is the output vector.

2.3.9. Batch Normalization

The BN mechanism is to stabilize the input distribution to a specific network layer during training (during a minibatch) [49]. The first two moments (mean and variance) of each activation's distribution are normalized to zero and one, respectively, by incorporating additional layers into the network. Following normalization, the batch-normalized inputs are typically scaled and shifted using trainable parameters to preserve the model's expressiveness. This normalization is applied before the preceding layer's non-linearity. The reduction of the so-called internal covariate shift (ICS) was one of the main driving forces behind the creation of Batch-Norm [49].

BN stabilizes the input distribution to a network layer during training by normalizing activations within each minibatch [49]. It standardizes the first two moments, mean x , and variance y_i , of each activation's distribution to zero mean and unit variance, then scales and shifts the normalized values using learnable parameters to retain model flexibility. This operation is applied before the layer's non-linearity and helps mitigate internal covariate shift, a key motivation behind Batch-Norm. The moments are mathematically expressed as given by

$$x = \frac{x_i - \mu}{\sqrt{\sigma^2 + \epsilon}}, \quad (8)$$

where x_i is the input activation value before normalization, μ is the mean of the batch (average of all x_i in the minibatch), σ^2 is the minibatch variance, and ϵ is a small constant needed for numerical stability (prevents division by zero).

$$y_i = \gamma x + \beta, \quad (9)$$

where γ is a learnable scaling parameter (restores the representation power of the network after normalization), and β is the learnable shifting parameter (adjusts the mean after normalization).

2.3.10. Adam Optimizer

The Adam optimizer efficiently computes adaptive learning rates by estimating the first and second moments of gradients, avoiding expensive higher-order computations [50]. Like momentum in SGD, Adam uses an exponentially decaying average of past gradients to stabilize updates and handle noisy gradients. It also maintains a decaying average of past squared gradients to scale learning rates per parameter, enabling larger updates for rare features and smaller ones for frequent ones. To counter initial bias toward zero, Adam includes a bias correction step for early iterations

$$\theta_{t+1} = \theta_t - \alpha \cdot \frac{m_t}{\sqrt{v_t + \epsilon}} \tag{10}$$

$$m_t = \frac{m_t}{1 - B_1^t}, v_t = \frac{v_t}{1 - B_2^t} \tag{11}$$

where θ_t is the model parameter at step t , α is the learning rate, controlling the step size, m_t is a bias-corrected first moment estimate, v_t is a bias-corrected second moment estimate, ϵ It is a small constant to prevent division by zero. B_1 and B_2 are decay rates for moving averages of gradients and squared gradients, typically set to 0.9 and 0.999, respectively.

2.4. TL-based DL Models

TL repurposes a model trained for one task to solve a different but related problem. By using the features it has already learned, this technique enhances performance and reduces training time. In our case, we aim to fine-tune 11 parameters, as outlined in Table 4, on our dataset and select the optimal one for the final training of the architecture.

Table 4. Hyper-parameters tuning, and the best of them is selected at which model gives the best results.

Parameter	Values under trial	Selected value
Learning rate	0.0001, 0.001, 0.01,0.1	0.001
Epochs	1,5,10,15,20,25,30,40,50	30
Minibatch (size)	32,64	64
Kernel depth	64	64
Kernel size	3 × 3	3 × 3
Activation	ReLU, Softmax	ReLU, Softmax
Pooling type	AvgPool, Max-Pool	AvgPool, Max-Pool
FC layers (fully connected)	1,2	1
Dropout	0.3, 0.4, 0.5	0.5
Solver name	ADaptive Moment estimation (ADAM)	ADAM

2.4.1. AlexNet

AlexNet is a groundbreaking image classification model that significantly advanced DL. Developed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton in 2012, it marked a major milestone by demonstrating the power of CNN and their broad applicability [51]. AlexNet accepts input images sized $227 \times 227 \times 3$ (RGB). Its first layer applies 96 kernels of size 11×11 with a stride of 4, followed by a ReLU activation and a MaxPooling operation. The second layer takes the output from the first layer and uses 256 kernels of size $5 \times 5 \times 48$ [52]. This layer utilizes 384 kernels of size $3 \times 3 \times 256$, without any pooling or normalization processes being applied in the third, fourth, and fifth layers. The fourth layer consists of 384 kernels measuring $3 \times 3 \times 192$. The fifth layer employs 256 kernels of size $3 \times 3 \times 192$. The fully connected layers each contain 4096 neurons. The network processes input images through convolution, activation, pooling, and fully connected layers to extract hierarchical features and classify them into categories. AlexNet’s use of GPU-based parallel processing for training was a breakthrough, drastically reducing computation time and allowing the model to be trained on the large-scale ImageNet dataset. It outpaced traditional methods by a wide margin and influencing subsequent CNN architectures like VGG, GoogLeNet, and ResNet. The model’s design demonstrated the practical feasibility of deep networks on complex real-world datasets, solidifying its place as a cornerstone in the development of modern computer vision models.

2.4.2. ResNet18

ResNet18 is a more compact variant of the ResNet50 model, featuring 18 layers that incorporate shortcut paths which help simplify the learning process in deep architectures and improve their ability to learn complex representations [53]. It uses shortcut paths that help maintain stability during the learning process of more complex model architectures, helping to alleviate the issue of vanishing gradients. With approximately 11 million parameters, it offers effective and robust architecture for image recognition applications. The key feature of ResNet18 is its use of residual connections, which allow the network to learn residual mappings instead of directly learning the desired output. This helps the model avoid performance degradation as the network depth increases.

2.4.3. ResNet50

ResNet50 is a 50-layer CNN that overcomes vanishing gradients through residual connections. These skip connections enable better gradient flow and allow the network to bypass layers, improving its ability to learn complex features. Key features of ResNet50 include BN, which stabilizes the learning process and accelerates convergence, and global average pooling, which reduces the number of parameters and mitigates overfitting. This architecture achieves outstanding performance on large-scale datasets like ImageNet and is widely used for tasks such as image classification, object detection, and TL, owing to its efficient design and ability to train deep models effectively [53].

2.4.4. SqueezeNet

SqueezeNet is a CNN designed to be 50 times smaller in terms of parameters than AlexNet, with a model size of less than 5MB [54]). It achieves this reduction through the use of fire modules, which are the core building blocks of the network. Each fire module contains a squeeze layer with 1x1 filters, followed by an expand layer that uses both 1x1 and 3x3 filters. This design allows for efficient feature extraction while minimizing computational cost. Additionally, deep compression techniques further reduce the model size to just 0.5MB, making it 510 times smaller than AlexNet. The architecture consists of approximately 18 layers, including one convolutional layer, a MaxPooling layer, eight fire modules, a global average pooling layer, and a softmax layer. Despite having far fewer parameters compared to traditional CNNs, SqueezeNet achieves competitive performance on benchmark tasks, making it ideal for resource-constrained applications such as mobile and embedded systems.

2.4.5. VGG16

VGG16 is a classic CNN architecture developed by Oxford's VGG group in 2014. With 16 layers (13 convolutional, 3 fully connected), it stacks small 3x3 filters to demonstrate the power of depth in image classification. The model employs MaxPooling layers after every few convolution layers to downsample the feature maps.

A key characteristic of VGG16 is its uniformity, where the same filter size (3x3) is used throughout the network, except for the fully connected layers. This uniform design makes the network easy to implement and replicate. The architecture is also recognized for demonstrating the importance of depth in DL models, influencing later architectures such as ResNet and GoogLeNet. However, due to its large number of parameters, VGG16 can be computationally expensive, making it less efficient for deployment in resource-constrained environments [55].

2.5 5-Fold CV

CV is used to evaluate the performance and generalization of a model. Initially, the entire dataset is divided into five equal groups called folds. For each iteration, one-fold is reserved as a testing set while the other four are used for training the model. The process is carried out five times, and each time a different fold is used as a validation set. After each iteration, different performance measures are used to evaluate the model. The average of five scores provides a more reliable estimate. 5-fold

CV eliminates the risk of over-fitting and ensures the generalization of the model to the new dataset. Figure 7 provides a more detailed picture of 5-fold CV.

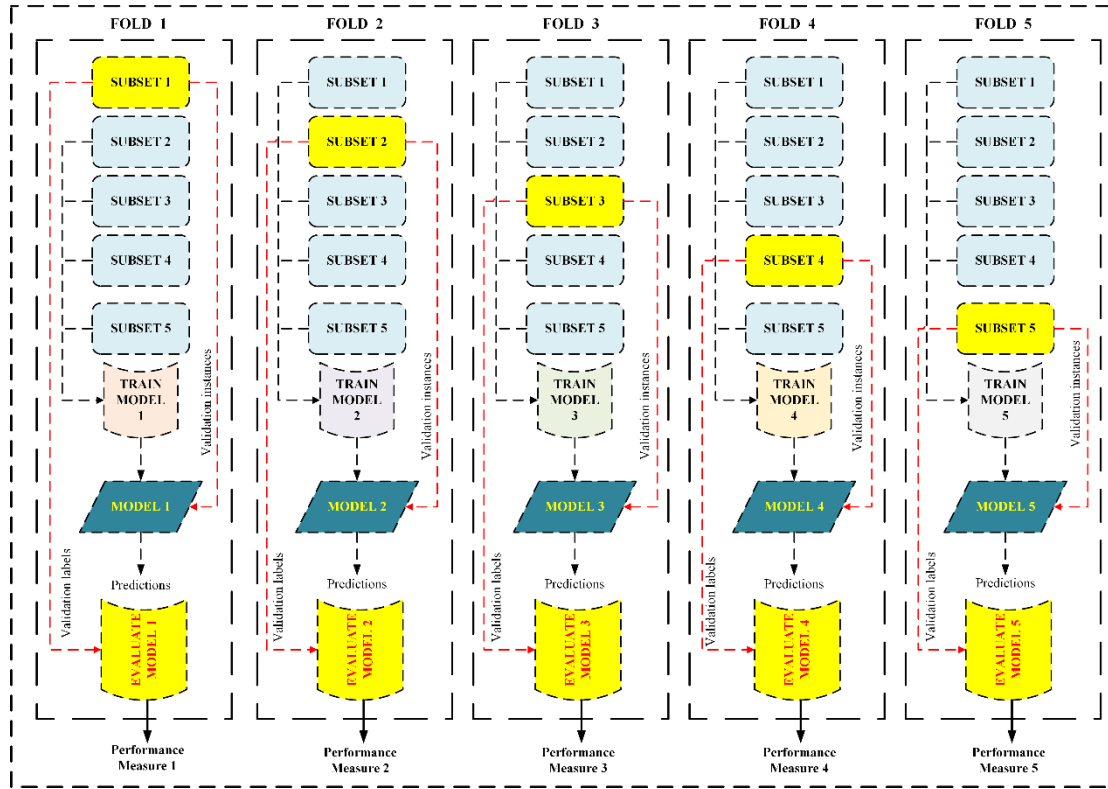


Figure 7. 5-fold CV which initially splits dataset into five equal parts. Then the process of training and testing is carried out 5 times. In each iteration, a different fold is crossed as testing, while the remaining four are assumed the training set.

2.6. Performance Measures

The performance metric is a type of evaluation that tells us about the performance of our model. It accomplishes two goals: techniques that do not work well can be abandoned, and those that appear to be promising in classifying our C19 dataset can be improved further. In supervised ML, we partition our data into 80.00% training and 20.00% test sets, utilizing the training data to train and validate the model, predict all instances of the test data, then compare the resultant predictions to the test set's actual values [56]. This allows us to assess if a new ML model's predictions outperform those of humans or pre-existing models in our test set.

2.6.1. Accuracy

Accuracy (A) metric is used to evaluate the performance of a DL classification model. It is given as the ratio of correct predictions to the total number of predictions made [57].

$$A = \frac{\text{number of correct prediction}}{\text{total number of prediction}} \quad (12)$$

2.6.2. Specificity

It is also referred to as the true negative rate. It represents the ability of the DL model to accurately predict the number of cases that do not have C19 or Pneumonia. When the model correctly predicts a diseased case, it is referred to as a true positive. When the model correctly predicts a healthy case, it is referred to as a true negative.

$$S_p = \frac{\text{true negative}}{\text{true negative} + \text{false positive}} \quad (13)$$

80% Sp of the model means it predicted that 80% of the tested cases lie in the N category, but it misclassified 20% cases [58].

2.6.3. Sn (Recall)

For multi-class classification tasks, the true positive rate, also referred to as Sn or recall, is measured per class to determine how accurately the model detects instances of that class. It is calculated as true positives divided by the sum of true positives and false negatives. The Sn formula is

$$Sn = \frac{\text{ture positive}}{\text{ture positive} + \text{false negative}} \quad (14)$$

A true positive refers to an instance that the model accurately identifies as positive, while a false negative is an actual positive incorrectly labeled as negative.[59].

2.6.4. Precision (Positive Predicted Value)

Precision indicates the accuracy of the model's positive predictions, calculated as the ratio of true positives to true positives plus false positives.[57].

$$Pr = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \quad (15)$$

2.6.5. Negative Predicted Value

Negative predictive value (NPV) measures the ability of a model to accurately determine true negatives or truly healthy instances [60]. It is given by

$$NPV = \frac{\text{true negative}}{\text{true negative} + \text{false negative}} \quad (16)$$

2.6.6. F1 – Score

F1-score combines precision and Sn/recall as it is the harmonic mean of both performance measures [61]. Its range lies between [0, 1]. The maximum value of the F1 score represents the better performance of the ML algorithm.

$$F1 - score = \frac{2 \times (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}} \quad (17)$$

F1-score is better to use when there are imbalanced classes in the dataset, and both false positives and false negatives are important to consider.

2.6.7. Area Under the Receiver Operating Characteristic ROC(AUC) curve

ROC(AUC) is a key measure for evaluating classification models over multiple threshold values [52]. It reflects the model's capability to separate classes across the full operating range. The ROC curve illustrates the balance between true positive and false positive rates, and the AUC expresses this separation as a single number. Higher ROC(AUC) scores mean stronger classification performance. A well-trained model will achieve a steeper ROC curve and a larger AUC, signifying better predictive ability.

2.6.8. Area Under the Precision Recall PR(AUC) curve

For datasets with significant class imbalance and few positive examples, PR(AUC) is more informative than the ROC(AUC) metric [52]. The PR(AUC), which plots precision against recall at various thresholds, highlights the balance between accuracy and Sn. It delivers a reliable assessment of performance in such challenging scenarios [60].

2.7. Optimized Weight Distribution Algorithm

An optimized weight distribution model can help mitigate bias in DL by dynamically adjusting weights based on model performance. This approach promotes fairness by assigning higher

importance to well-performing components, while penalizing those that underperform. This ensemble strategy for performance optimization improves robustness and generalization, ensuring more reliable predictions.

The pseudocode in Figure 8 describes an ensemble contribution aggregation process for four performance metrics from five potential DL models to create an ensemble performance representation. First, four key metrics *A*, F1 score, ROC (AUC), and PR (AUC), are calculated for the five DL models. The procedure aggregates performance from five architectures (Model_1 ... Model_5). It first normalizes each metric across models to get weights, scales each model's metric by its weight to compute an "individual contribution," and then sums those contributions to produce final aggregated scores.

```

Begin

// Step 1: Input performance measures of 5 architectures (Model_1 to Model_5)
// Input: Acc_1, F1_1, ROC_1, PR_1 // for Model_1
//       Acc_2, F1_2, ROC_2, PR_2 // for Model_2
//       Acc_3, F1_3, ROC_3, PR_3 // for Model_3
//       Acc_4, F1_4, ROC_4, PR_4 // for Model_4
//       Acc_5, F1_5, ROC_5, PR_5 // for Model_5

// Step 2: Calculate weights for each architecture
For i = 1 to 5:
    Acc_Weight_i = Acc_i / (Acc_1 + Acc_2 + Acc_3 + Acc_4 + Acc_5)
    F1_Weight_i = F1_i / (F1_1 + F1_2 + F1_3 + F1_4 + F1_5)
    ROC_Weight_i = ROC_i / (ROC_1 + ROC_2 + ROC_3 + ROC_4 + ROC_5)
    PR_Weight_i = PR_i / (PR_1 + PR_2 + PR_3 + PR_4 + PR_5)

// Step 3: Derivation of individual contribution
For i = 1 to 5:
    New_Acc_Model_i = (Acc_Weight_i * Acc_Model_i)
    New_F1_Model_i = (F1_Weight_i * F1_Model_i)
    New_ROC_Model_i = (ROC_Weight_i * ROC_Model_i)
    New_PR_Model_i = (PR_Weight_i * PR_Model_i)

// Step 4: Partial contributions summation for the respective models
For i = 1 to 5:
    Acc = 0; Acc = New_Acc_Model_i + Acc
    F1 = 0; F1 = New_F1_Model_i + F1
    ROC = 0; ROC = New_ROC_Model_i + ROC
    PR = 0; PR = New_PR_Model_i + PR

End

```

Figure 8. Optimized weighted algorithm for the ensemble contribution aggregation using performance measures for multiclass C19 dataset.

3. Results and Discussion

The experiments were conducted on a Lenovo ThinkPad running Microsoft Windows 11 Pro (Build 26100) having an Intel® Core™ i7 processor (~2.59 GHz), 64 GB RAM, on an x64-based architecture, and a GPU is an NVIDIA Quadro P3200 with Max-Q Design (6 GB dedicated VRAM, 1792 CUDA cores).

In this study, we implemented evaluation of five widely used TL models, namely ResNet50, VGG16, AlexNet, SqueezeNet, and ResNet18, for the classification of C19 from chest radiographs [62]. A 5-fold CV procedure was applied for training and validating purpose, ensuring stable results. The hyper-parameters such as batch size, learning rate, number of epochs, and optimizer selection were kept the same for all models run to ensure a fair comparison. The training process was

monitored to prevent over-fitting and reach optimal performance levels on both training and validation sets.

3.1. AlexNet

In this section, a comprehensive evaluation of the classification capability of a pre-trained AlexNet model is conducted for identifying C19, LO, and N cases on chest X-rays. Table 5 illustrates the results, showing the sensitivity analysis of the model’s performance across different epochs. There was improvement in performance scores with increase in epochs across all metrics. Sn increased from 90.39% to 95.23% , and Sp rose from 79.76% to 90.94% with epochs rising in the range [1,50]. The Pr trend demonstrated a steady improvement, increasing from 81.05% to 92.02% rising epochs in the range [1,50], with the NPV progressed from 90.54% to 92.02%. The ROC(AUC) values increased from 0.94 at to 0.95 by increasing epochs. A similar trend was observed in the PR(AUC) values, improving from 0.89 to 0.93, indicating a better performance. The F1-score rose significantly from 79.42% to 91.44%, highlighting the model’s improved performance. The overall A of the model showed growth from 81.93% to 91.53% by increasing epochs from 1 to 50.

Table 5. Classification performance of the pre-trained AlexNet model on the radiography-preprocessed dataset (dataset classes: C19, LO, and N; epochs represented by Es; and 5-fold CV).

Es	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
1	90.39±0.72	79.76±2.90	81.05±2.16	90.54±0.77	0.94±0.00	0.89±0.01	79.42±1.24	81.93±1.11
5	92.65±0.55	85.40±1.45	85.70±2.20	92.73±0.90	0.96±0.00	0.93±0.00	85.45±1.33	86.12±1.30
10	94.19±0.26	88.83±0.81	88.95±0.32	94.27±0.20	0.97±0.00	0.95±0.00	88.85±0.50	89.49±0.40
15	94.55±0.32	89.40±0.70	90.75±0.50	94.80±0.30	0.97±0.00	0.95±0.00	89.95±0.57	90.40±0.52
20	94.44±0.25	88.96±0.95	90.62±0.32	94.74±0.20	0.97±0.00	0.95±0.00	89.68±0.61	90.21±0.42
25	95.05±0.23	90.43±0.50	91.77±0.48	95.27±0.25	0.97±0.00	0.96±0.00	91.04±0.44	91.29±0.41
30	95.23±0.29	90.89±0.64	92.00±0.50	95.42±0.148	0.97±0.0	0.96±0.00	91.38±0.53	91.60±0.39
40	95.21±0.17	90.85±0.39	91.88±0.30	95.36±0.16	0.97±0.00	0.96±0.00	91.33±0.35	91.53±0.30
50	95.23±0.17	90.94±0.49	92.02±0.23	95.37±0.11	0.97±0.00	0.96±0.00	91.44±0.33	91.57±0.25

In Figure 9, ROC curves are illustrated for the ResNet18 model. The ROC curve offers insight into detection trend versus false alarm rates. The ROC(AUC) value reflects the model’s overall classification skill, ranging from 1.0 (perfect) to 0.5 (none).

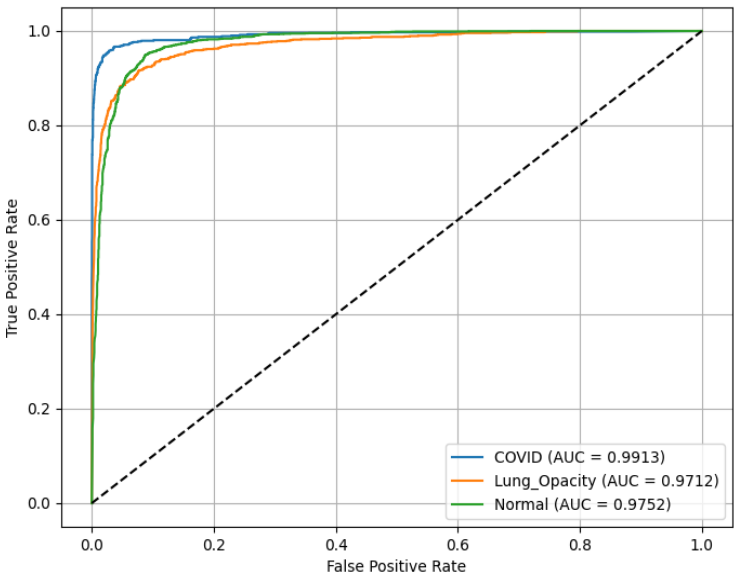


Figure 9. ROC curves for multiclass classification using pre-trained AlexNet classifier (classes: C19, LO, and N; chest X-ray preprocessed dataset; 5-fold CV).

AlexNet resulted in excellent performance across all classes as illustrated in Figure 10, with PR(AUC) values as follows: C19 (blue curve) achieves 0.97, LO (orange curve) reaches 0.95, reflecting high reliability in LO detection, and N (green curve) attains 0.97.

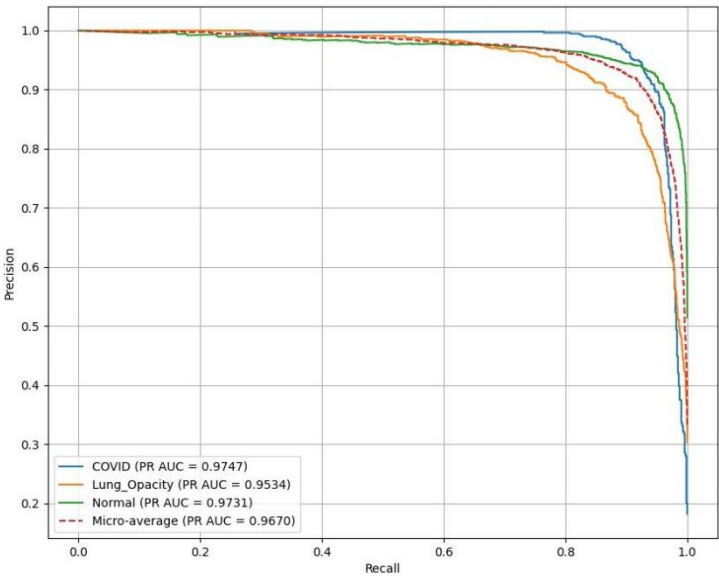


Figure 10. PR curves for multiclass classification using pre-trained AlexNet classifier (classes are C19, LO, and N in chest X-ray preprocessed dataset using 5-fold CV).

The confusion matrix for this experimentation with pre-trained AlexNet model is illustrated in Figure 11. The model accurately predicted 653 C19 cases, with some misclassifications: 37 instances were identified as LO, and 33 as N. For the LO class, the model correctly identified 1063 cases, though 12 were misclassified as C19, and 127 as N. For the N class, the model achieved a high accuracy, correctly identifying 1904 cases, with only 15 misclassifications as C19 but 120 as LO. The color scale is a visual depiction of the occurrence of each category, with darker shades pointing higher values. Overall, the model demonstrates appropriate performance, particularly in distinguishing N from C19, but there is some feature similarity existing between LO and the remaining classes leading to lagging results.

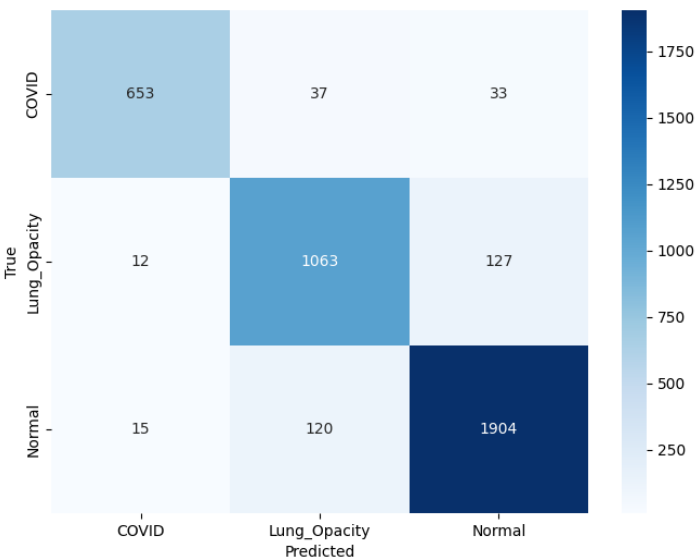


Figure 11. Confusion matrix for multiclass classification using pre-trained AlexNet classifier (the best fold result with classes as C19, LO, and N defined in the preprocessed dataset using 5-fold CV).

3.2. ResNet18

The details related to the classification performance of a pre-trained ResNet18 model are discussed in this section under identical conditions defined in Section 3. The classification performance of the ResNet18 model is illustrated in Table 6. It is clear that as the number of epochs increases, the model’s performance improves. For example, at epoch 1, the *A* is 85.76±1.58%, while at epoch 40, it reaches 92.74±0.50%. Specific metrics show consistent improvements, with *Sp* rising from 92.50±0.29% to 95.67±0.15% and *Sn* rising from 85.12±1.24% to 92.40±0.30% with epochs increasing in the range [1,40]. The model’s AUC for both ROC and PR curves follows a similar trend. This indicates that as the model trains over more epochs, it becomes more reliable in distinguishing between the various classes. Beyond epoch 40, overfitting is expected to impact generalization due to training on image noise, which hinders learning and affects testing results.

Table 6. Classification performance of the pre-trained ResNet18 model on the radiography-preprocessed dataset (dataset classes: C19, LO, and N; epochs represented by Es; 5-fold CV).

Es	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
1	92.50±0.29	85.12±1.24	85.21±3.10	92.40±1.13	0.96±0.00	0.93±0.00	84.57±1.33	85.76±1.58
5	94.48±0.66	89.44±1.21	90.30±1.82	94.70±0.98	0.97±0.00	0.96±0.00	89.71±1.37	90.12±1.59
10	94.78±1.03	90.45±1.77	90.99±1.77	94.80±1.36	0.98±0.00	0.96±0.00	90.53±2.07	90.51±2.41
15	95.10±0.65	91.20±1.00	91.60±1.20	95.30±0.80	0.98±0.00	0.97±0.00	91.00±1.50	91.10±1.20
20	95.35±0.40	91.70±0.70	91.95±0.70	95.45±0.50	0.98±0.00	0.97±0.00	91.70±1.00	91.60±0.90
25	95.57±0.15	92.12±0.41	92.32±0.60	95.61±0.25	0.98±0.00	0.97±0.00	92.19±0.20	92.14±0.26
30	95.74±0.37	92.28±0.86	92.92±0.54	95.98±0.19	0.98±0.00	0.97±0.00	92.56±0.57	92.62±0.49
40	95.67±0.15	92.40±0.30	92.87±0.64	95.81±0.42	0.98±0.00	0.97±0.00	92.59±0.39	92.74±0.50
50	95.64±0.14	92.10±0.59	92.82±0.56	95.68±0.24	0.98±0.00	0.97±0.00	92.41±0.32	92.28±0.31

ResNet18 results are presented in Figure 12, demonstrating discriminative ability across all classes. The ROC(AUC) values are as follows: C19 (blue curve) achieves an ROC(AUC) of 0.99, indicating excellent performance in identifying C19 cases. LO (orange curve) has an AUC of 0.97, reflecting high accuracy in detecting LO; and N (green curve) records an AUC of 0.98, showing robust classification of N cases. The diagonal line (dashed) represents a random classifier (AUC = 0.5) for

reference, highlighting the model’s superior performance well above chance level across all categories.

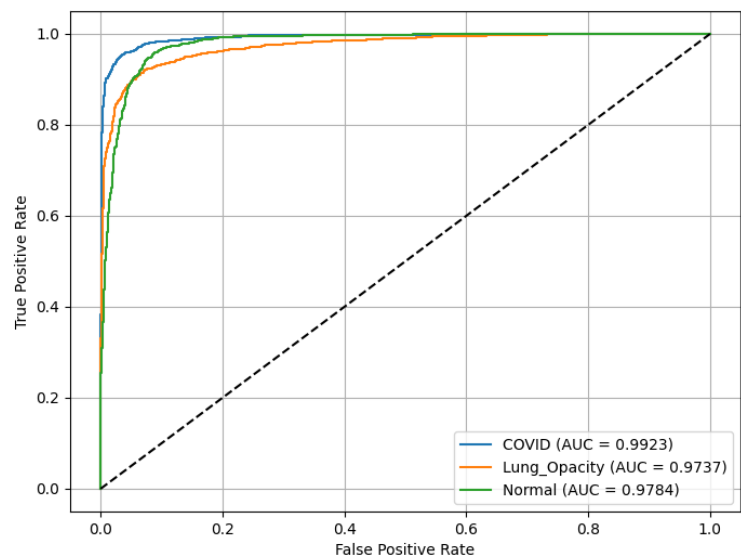


Figure 12. ROC curves for multiclass classification using pre-trained ResNet18 (classes: C19, LO, and N in chest X-ray preprocessed dataset using 5-fold CV).

PR(AUC) is illustrated in Figure 13, showing the performance of a ResNet18 model. C19 achieves a PR(AUC) of 0.98, LO has an AUC of 0.96, reflecting high reliability in identifying LO, and N (green curve) records an AUC of 0.97.

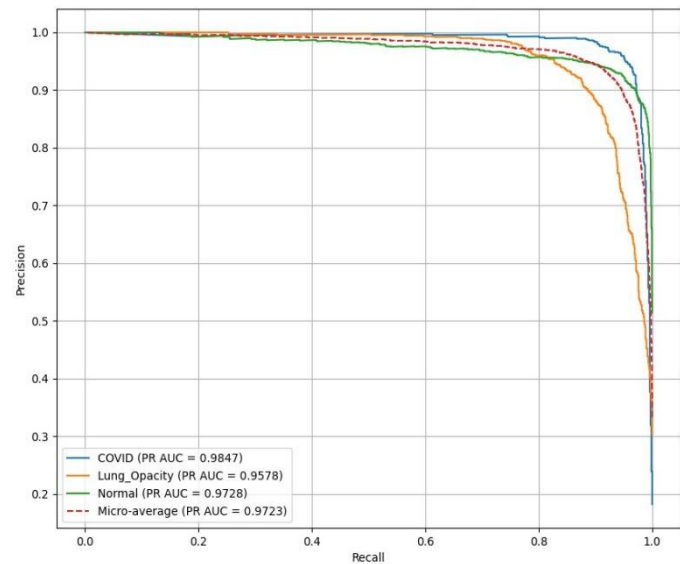


Figure 13. PR curves for multiclass classification using pre-trained ResNet18 model (classes: C19, LO, and N present in chest X-ray preprocessed dataset, and employing 5-fold CV).

The model correctly classified 687 C19 cases, with 16 misclassified as LO and 20 as N, as illustrated in the confusion matrix shown in Figure 14. For LO, 1083 cases were identified correctly, with 10 misclassified as C19 and 109 as N. For N cases, 1894 was correctly classified, with 14 misclassified as C19 and 131 as LO. The color intensity, ranging from light to dark blue, represents the number of instances, with the scale on the right indicating counting from 0 to 1750.

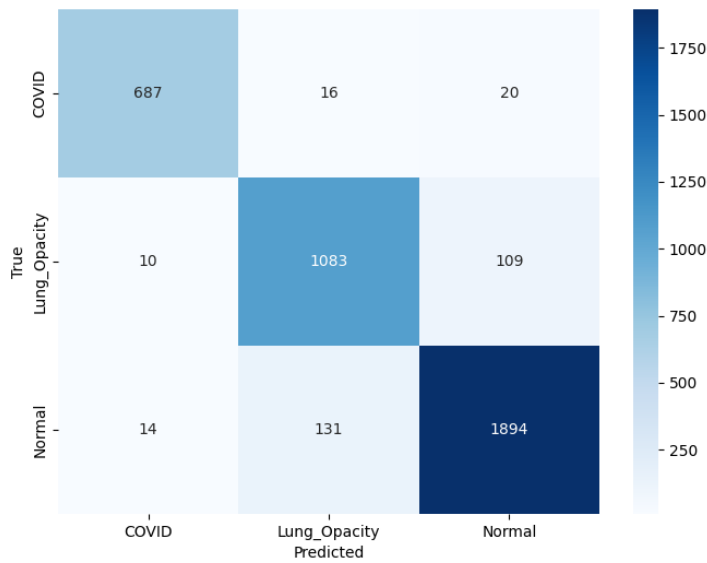


Figure 14. Confusion matrix for multiclass classification using pre-trained ResNet18 model (with classes C19, LO, and N in the chest X-ray preprocessed dataset using 5-fold CV).

3.3. ResNet50

The classification performance of the pre-trained ResNet50 model for identifying C19, LO, and N is illustrated in Table 7. As the number of epochs increased, there was a consistent improvement in the model’s performance across various metrics. The Sp risen from 92.43% at epoch 1 to 96.24% at epoch 30, indicating a substantial improvement in the model’s ability to identify positive cases correctly until the overfitting threshold is reached. Similarly, Sp (Rc/Sn) rose from 85.55% at epoch 1 to 93.24% at epoch 30, beyond which a slight decline in the performance scores is observed. Other metrics followed the same trend. The ROC(AUC) remained high throughout, reaching 0.98 at epoch 50. Similarly, the PR(AUC) remained strong, improving from 0.94 at epoch 1 to 0.97 at epoch 50, indicating its capability to handle imbalanced classes. The F1-score steadily enhanced from 84.30% at epoch 1 to 93.44% at epoch 30. Finally, A showed a rise from 85.17% at epoch 1 to 93.41% at epoch 30, reflecting the overall enhancement in model performance.

Table 7. Classification performance of the pre-trained ResNet50 model on the radiography-preprocessed dataset (dataset classes: C19, LO, and N; epochs represented by Es; and 5-fold CV).

Es	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
1	92.43±1.52	85.55±2.60	84.81±4.08	92.04±2.18	0.96±0.00	0.94±0.00	84.30±4.33	85.17±4.28
5	94.76±0.15	90.00±0.43	90.18±1.38	94.80±0.55	0.98±0.00	0.96±0.00	89.94±0.61	90.44±0.61
10	95.28±0.26	90.89±0.93	92.40±0.45	95.75±0.18	0.98±0.00	0.96±0.00	91.54±0.44	91.92±0.29
15	95.75±0.21	92.03±0.59	92.88±0.21	95.95±0.07	0.98±0.00	0.97±0.00	92.43±0.38	92.57±0.25
20	95.98±0.13	92.69±0.32	93.25±0.70	96.21±0.31	0.98±0.00	0.97±0.00	92.93±0.21	93.00±0.26
30	96.24±0.09	93.24±0.30	93.66±0.25	96.36±0.10	0.98±0.00	0.97±0.00	93.44±0.13	93.41±0.12
40	96.06±0.19	92.96±0.54	93.48±0.20	96.21±0.11	0.98±0.00	0.97±0.00	93.21±0.34	93.13±0.27
50	95.99±0.18	92.77±0.60	93.48±0.33	96.16±0.12	0.98±0.00	0.97±0.00	93.11±0.44	93.04±0.29

This ROC curve illustrates the performance of the ResNet50 model in classifying three chest X-ray classes as illustrated in Figure 15: C19, LO, and N, based on test data from Fold 2. The tight clustering of these ROC curves near the top-left corner of the plot highlights ResNet50’s high capability in minimizing false positives while maximizing true positives across all three classes.

Figure 16 shows that C19 achieves the highest PR(AUC) of 0.99 at varying classification thresholds. The N class follows with a strong PR(AUC) of 0.98, while LO records a slightly lower AUC of 0.96, suggesting somewhat greater difficulty in distinguishing it from other conditions.

Figure 17 presents a multiclass confusion matrix for the classification of chest X-ray images into three classes using a pre-trained ResNet50 model. For C19, 688 predictions were correct, while 17 were LO and 18 N. For LO, 1090 were accurate, with 12 predicted as C19 and 100 as N. For N, the model correctly classified 1911, misclassifying 117 as LO and 11 as C19. The relatively low number of misclassifications indicates that the model is performing well, although some overlap remains, particularly between LO and N cases.

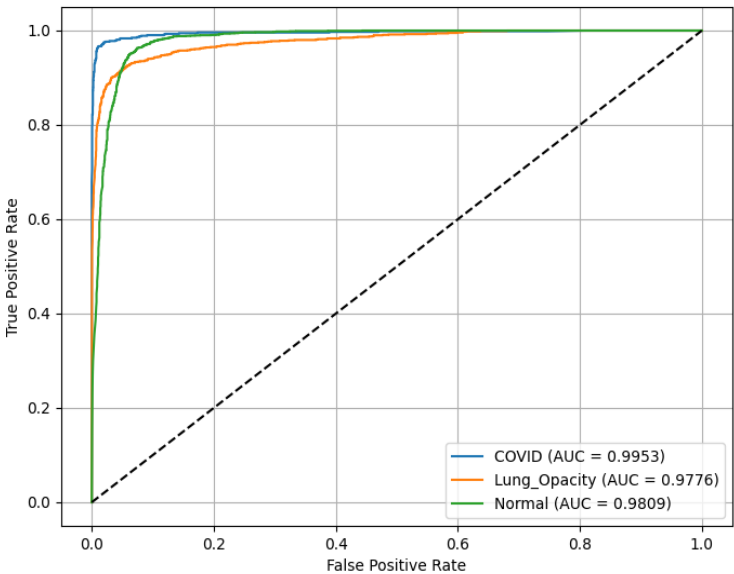


Figure 15. ROC curves for multiclass classification using pre-trained ResNet50 architecture (classes: C19, LO, and N for chest X-ray preprocessed dataset using 5-fold CV).

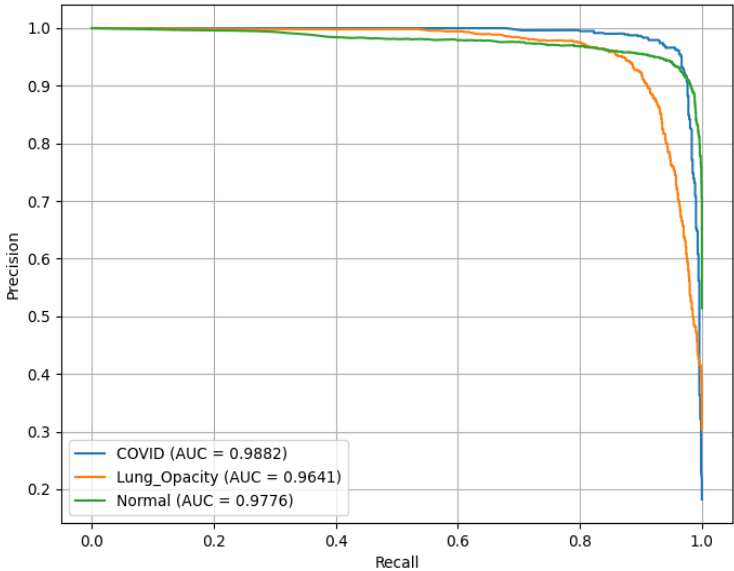


Figure 16. PR curves for multiclass classification using pre-trained ResNet50 (classes: C19, LO, and N in chest X-ray preprocessed dataset using 5-fold CV).

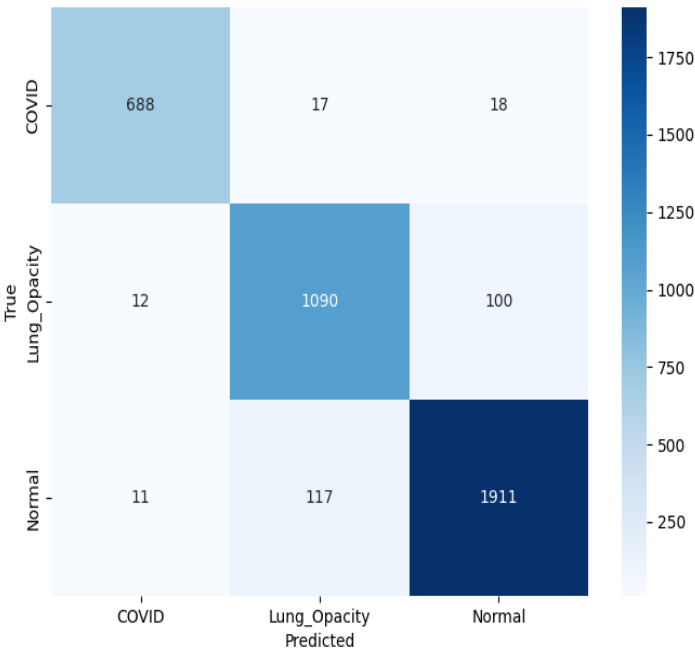


Figure 17. Confusion matrix for multiclass classification using pre-trained ResNet50 (classes: C19, LO, and N; Dataset: chest X-ray preprocessed dataset with 5-fold CV).

3.4. SqueezeNet

The classification performance results of the pre-trained SqueezeNet are illustrated in Table 8. For this experimentation, the results have been shown for epochs 20, 25, 30, 40, and 50. The performance metrics indicate that the SqueezeNet model achieved its best results at epoch 25. Specifically, the Sp reached 96.52%, the highest value observed across all epochs, demonstrating the model’s excellent ability to identify true positives correctly. The Sn secured 93.95%, and the Pr was 94.65%, reflecting reliable predictions for positive cases. The NPV at epoch 25 was 96.71%, further highlighting the model’s ability to predict true negatives accurately. In terms of area under the curve (AUC), the model achieved an ROC(AUC) of 98.94% and a PR(AUC) of 98.36%, both indicating excellent performance. The F1-score was also high at 94.01%, balancing both precision and recall effectively. The A was 94.01%, again showing that the model’s overall performance was powerful at epoch 25.

Table 8. Classification performance of the pre-trained SqueezeNet model on the radiography-preprocessed dataset (dataset classes: C19, LO, and N; epochs represented by Es; and 5-fold CV).

Es	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
20	96.25±0.88	93.41±1.44	94.26±2.08	96.51±1.3	98.9±0.07	98.31±0.17	93.49±2.01	93.48±2.10
25	96.52±0.37	93.95±0.72	94.65±0.9	96.71±0.59	98.94±0.13	98.36±0.21	94.01±0.81	94.01±0.85
30	96.63±0.6	94.18±1.04	93.97±2.73	96.64±1.04	98.91±0.15	98.32±0.24	93.95±1.65	93.95±1.69
40	95.98±1.3	92.18±3.72	94.57±1.21	96.65±0.75	98.77±0.36	98.12±0.51	93.32±2.08	93.39±1.98
50	90.78±0.43	80.04±0.76	81.00±1.23	90.53±0.66	0.94±0.00	0.89±0.00	80.03±1.04	82.24±1.22

Figure 18 represents ROC(AUC) curves asses the performance of a SqueezeNet model. LO has ROC(AUC) of 0.94, C19 achieves ROC(AUC) of 0.94, reflecting high reliability in identifying LO, and N (green curve) records ROC(AUC) of 0.9493.

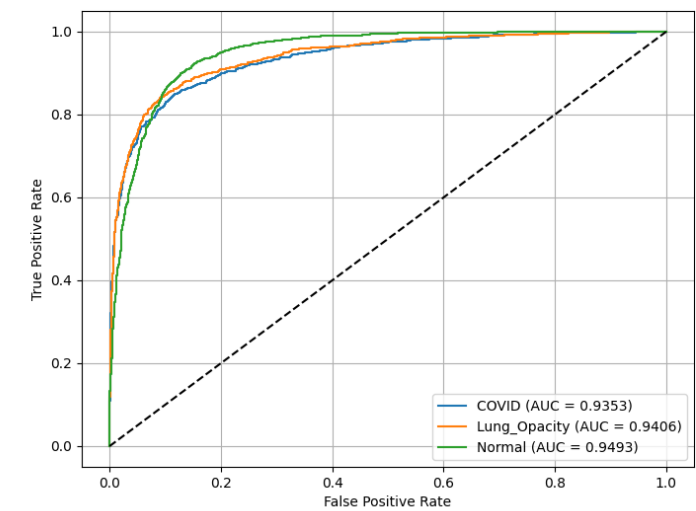


Figure 18. ROC curves for multiclass classification using pre-trained SqueezeNet framework (classes as C19, LO, and N in chest X-ray preprocessed dataset; 5-fold CV).

Figure 19 presents the PR(AUC) for the SqueezeNet model used to classify chest X-ray images into three classes. C19 (blue curve) achieves PR(AUC) of 0.85. LO (orange curve) has PR(AUC) of 0.89, reflecting high reliability in identifying LO; and N (green curve) records PR(AUC) of 0.92.

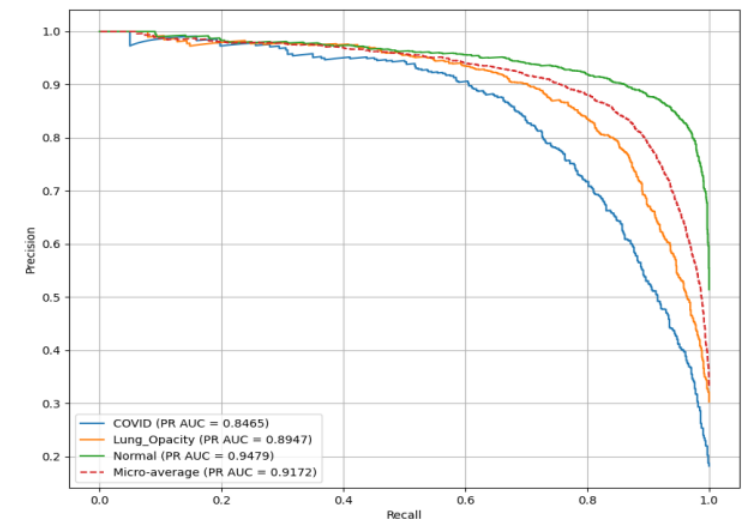


Figure 19. PR curves for multiclass classification using pre-trained SqueezeNet model (classes: C19, LO, and N forming the chest X-ray preprocessed dataset; 5-fold CV).

In Figure 20 illustrating the confusion matrix for SqueezeNet model, C19 class, out of the total actual cases, 494 were correctly classified, while 153 were misclassified as LO and 76 as N. It indicates considerable confusion particularly with LO cases. In the LO class, the model performs more robustly, with 1051 correct predictions, while 36 were incorrectly labeled as C19 and 115 as N, showing relatively balanced misclassification. In the N class, the model performs more robustly, with 1737 correct predictions, while 79 were incorrectly labeled as C19 and 223 as LO.

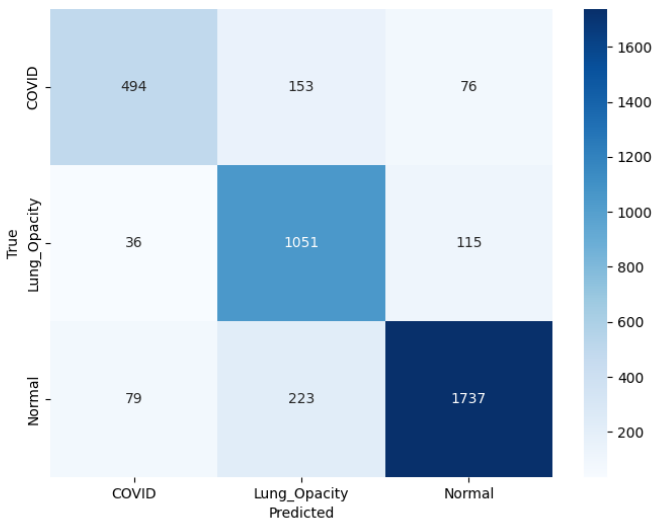


Figure 20. Confusion matrix for multiclass classification using pre-trained SqueezeNet model (classes: C19, LO, and N forming chest X-ray preprocessed dataset; 5-fold CV).

For visual analysis, three chest X-ray classes using high-dimensional features extracted from the SqueezeNet model are illustrated in Figure 21. For this, t-SNE, a non-linear dimensionality reduction method, projects feature vectors into a 2D space, thereby improving the representation by minimizing crowding and revealing class clusters. As shown, the model learns discriminative features where C19 samples tend to form a relatively compact cluster. At the same time, some overlap is observed between LO and N categories due to inherent similarities in chest radiographs. The C19 cluster remains discriminative, highlighting that the results for this class are more impressive than those of other options.

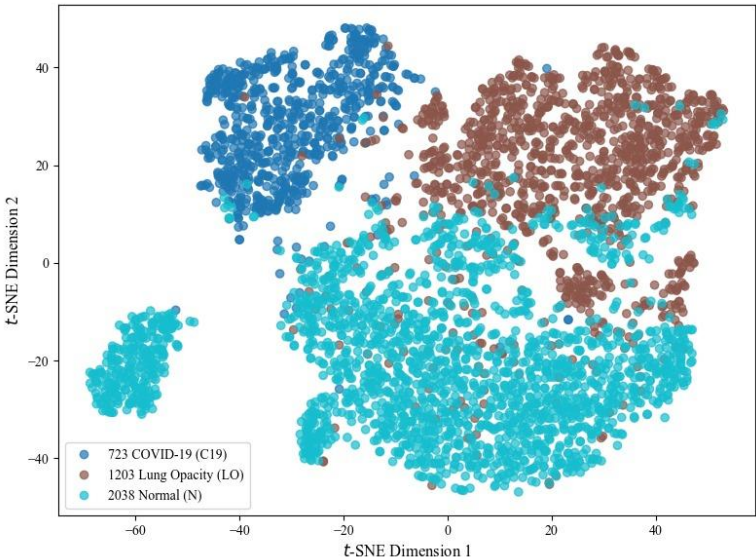


Figure 21. t-SNE plot showing the feature distribution of C19 (723 instances in blue), LO (1203 instances in brown) and N (2038 instances in teal) classes run during the testing phase.

3.5. VGG16

A detailed experimentation of the classification performance of the pre-trained VGG16 model (under identical conditions) is illustrated in Table 9. At epoch 1, the Sp was 90.43%, with the Sn at 79.04%, and Pr at 81.36%. The NPV was 91.47%, and the ROC(AUC) was 0.93, indicating the model’s

initial classification capabilities. As the number of epochs increased, the model showed steady improvements across all metrics. By epoch 25, the Sp reached 93.05%, the Specificity improved to 83.50%, and the F1-score reached 85.92%. At epoch 50, the Sn peaked at 94.83%, while the Specificity was the highest at 89.11%. The ROC(AUC) reached 0.98 at epoch 50, indicating excellent performance in distinguishing between the three classes. The model's A improved from 83.12% at epoch 1 to 90.26% at epoch 50, reflecting overall enhancement in classification. PR(AUC) reached 0.96 at epoch 50. The F1-score steadily improved, particularly at higher epochs, culminating at 90.26% in epoch 50, showing a balanced performance between precision and recall.

Table 9. Classification performance of the pre-trained VGG16 model on the radiography-preprocessed dataset (dataset classes: C19, LO, and N; epochs represented by Es; and 5-fold CV).

Es	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
1	90.43±1.25	79.04±3.99	81.36±0.89	91.47±0.89	0.93±0.02	0.93±0.12	82.12±2.99	83.12±1.70
5	91.43±1.14	81.04±3.67	86.36±0.81	92.41±0.85	0.96±0.01	0.92±0.01	82.37±2.96	84.87±1.70
10	92.10±0.90	82.50±2.80	87.50±0.70	93.00±0.80	0.96±0.00	0.93±0.01	83.90±2.20	86.00±1.30
15	92.75±0.65	83.80±2.00	88.25±0.60	93.60±0.60	0.96±0.00	0.93±0.01	85.25±1.80	87.20±1.00
20	93.00±0.40	84.60±1.30	89.00±0.40	93.85±0.45	0.96±0.00	0.94±0.00	86.20±1.20	87.90±0.70
25	93.05±0.35	85.00±1.10	89.40±0.30	93.90±0.35	0.96±0.00	0.94±0.00	86.60±1.00	88.10±0.60
30	93.11±0.30	85.40±0.98	89.79±0.19	94.02±0.27	0.97±0.00	0.94±0.00	87.14±0.76	88.33±0.53
40	94.62±0.11	88.76±0.47	91.63±0.25	95.27±0.09	0.97±0.00	0.95±0.00	90.01±0.24	90.82±0.14
50	94.83±0.22	89.11±0.69	91.77±0.33	95.34±0.16	0.98±0.00	0.96±0.00	90.26±0.54	91.04±0.37

Figure 22 represents PR(AUC) curves that illustrates the performance of the VGG16 model. C19 and LO each achieved an AUC of 0.93, while N (green curve) achieved 0.96.

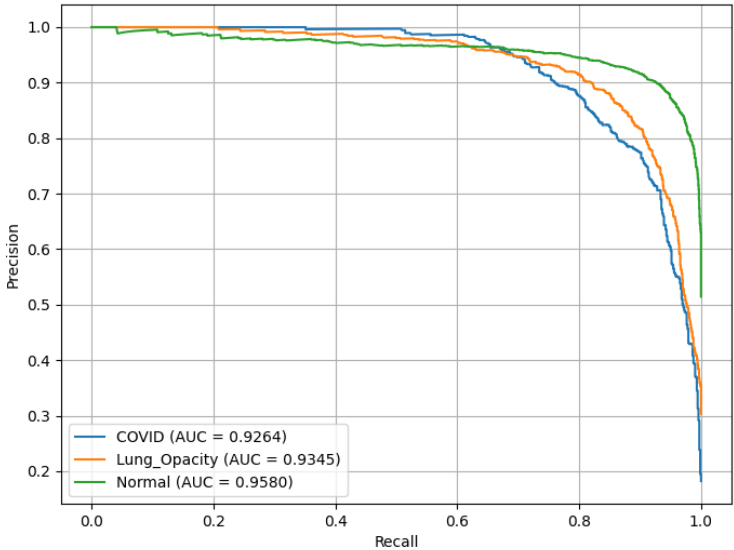


Figure 22. PR curves for multiclass classification using pre-trained VGG16 model (classes: C19, LO, and N forming chest X-ray preprocessed dataset; 5-fold CV).

Figure 23 shows that VGG16 achieved an ROC(AUC) of 0.99 for the C19 class, 0.97 for the LO class, and 0.97 for the N class, indicating excellent classification performance. The curves' positions near the top-left corner of the plot reflect the model's high Sn and low false positive rate, particularly for the C19 class.

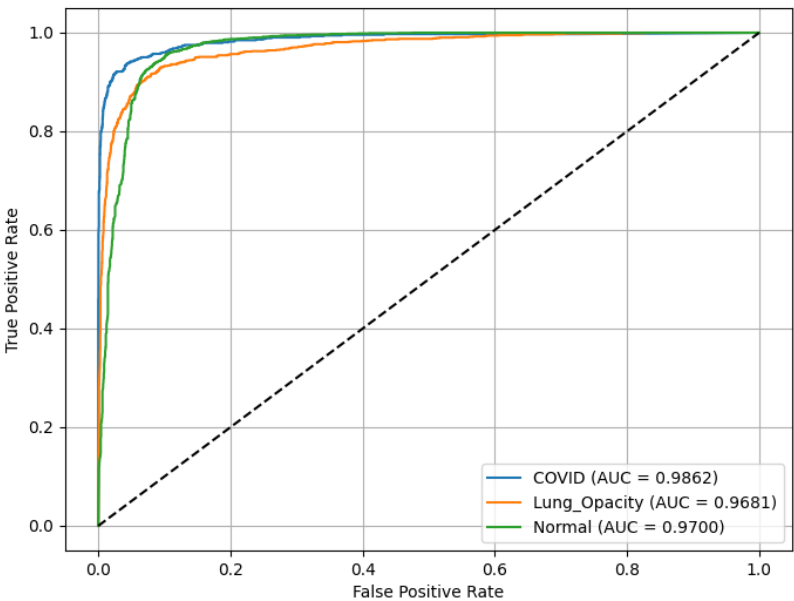


Figure 23. ROC curves for multiclass classification using pre-trained VGG classifier (classes: C19, LO, and N forming the chest X-ray preprocessed dataset; 5-fold CV).

Figure 24 shows that VGG16 correctly predicted 629 C19, 1,075 LO, and 1,934 N cases. Errors included 53 C19 mislabeled as LO and 41 as N. Among LO samples, 5 were misclassified as C19 and 122 as N. Among N samples, 18 were predicted as C19 and 87 as LO.

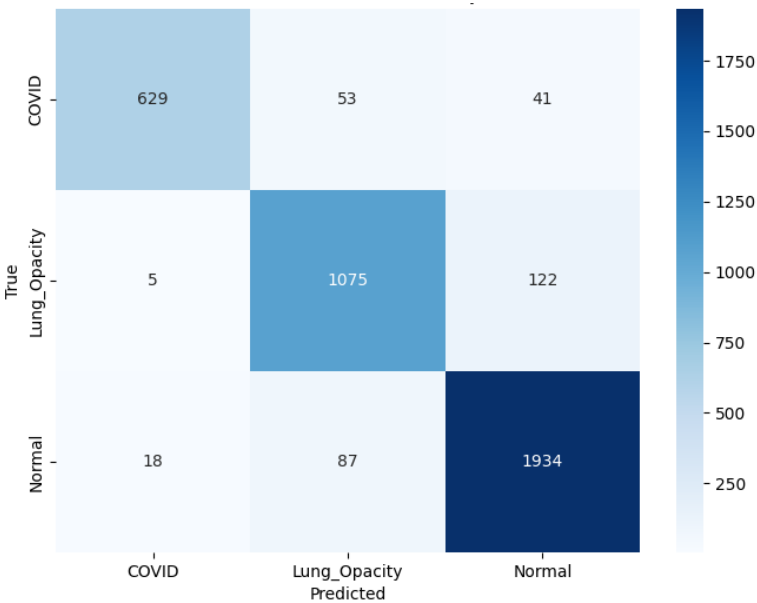


Figure 24. Confusion matrix for multiclass classification using pre-trained VGG16 model (for classes C19, LO, and N defining the chest X-ray preprocessed dataset using 5-fold CV).

3.6. Ensemble Contribution Aggregation Results

The integration of five DL algorithms, ResNet50, ResNet18, AlexNet, SqueezeNet, and VGG16, provides valuable insights into their ensemble performance across various diagnostic metrics. Table 10 illustrates a comparison of the performance of five pretrained DL models. The comparison is based on various performance metrics as illustrated in Section 2.6.

In terms of *A*, SqueezeNet outperforms all other models with the highest accuracy of 94.01%, closely followed by ResNet50 at 93.41%. ResNet18 achieves a slightly lower accuracy of 92.74%, while AlexNet scores 91.60%. This indicates that SqueezeNet leads in terms of overall performance in this category. For recall (Rc/Sn), SqueezeNet again takes the lead with a recall score of 93.95%, showing its ability to identify true positives. ResNet50 and ResNet18 also perform strongly with recall values of 93.24% and 92.42%, respectively. Regarding Pr, SqueezeNet stands out with a precision score of 94.65%, followed by ResNet50 at 93.66%. The other models, AlexNet, ResNet18, and VGG16, show slightly lower Pr, indicating SqueezeNet’s superior ability to identify true positive results with fewer false positives.

For ROC(AUC) and PR(AUC), SqueezeNet remains the top performer, achieving an AUC score of 0.98 in both ROC and PR curves. ResNet50 and ResNet18 also perform well, each scoring 0.98 in ROC(AUC) and 0.97 in PR(AUC), reflecting their ability differentiating positive/ negative cases. In terms of the F1-score SqueezeNet again leads with a performance score of 94.01%, followed by ResNet50 with 93.44%. VGG16 achieves the lowest F1-score at 90.26%, indicating it may struggle with balancing Pr (91.77%) and recall (89.11%) in comparison to other models. The accuracy shows that SqueezeNet is the most effective model across the various metrics, with ResNet50 follows closely in second place, showing competing overall performance.

Table 10. Comparison of performance metrics for various deep learning models (AlexNet, ResNet18, ResNet50, SqueezeNet, and VGG16) on the preprocessed COVID-19 chest X-ray dataset(using 5-fold cross-validation; the best results for each of the architectures.).

Method	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
AlexNet	95.23	90.89	92.00	95.42	0.97	0.96	91.38	91.60
ResNet18	95.67	92.42	92.87	95.81	0.98	0.97	92.59	92.74
ResNet50	96.24	93.24	93.66	96.36	0.98	0.97	93.44	93.41
SqueezeNet	96.52	93.95	94.65	96.71	0.98	0.98	94.01	94.01
VGG16	94.83	89.11	91.77	95.34	0.98	0.96	90.26	91.04

For the weighted-ensemble performance, Table 11 shows the contribution of the weighted metrics for the five CNN backbone models and the resulting weighted totals. Weights are consistent across models (around 18-20% each), with slight increases for SqueezeNet and ResNet50 in all columns, suggesting complementary contributions rather than reliance on a single backbone. The optimized ensemble achieves strong overall performance: 92.57% accuracy, 92.36% F1-score, 93.00% Precision, and 91.95% Recall. Discrimination is excellent, with an ROC(AUC) of 0.98 and PR(AUC) of 0.97, which is impressive for a three-class imbalanced dataset. Summing up, combining the five models with an optimized weight distribution results in a balanced classifier that provides robustly optimized outcomes by finely balancing the contribution from each classifier.

A visual comparison of various model evaluation scores, including PPV, NPV, F1-score, and *A*, for different studies is illustrated in Figure 25. These models are represented as bars on the x-axis: AlexNet, ResNet18, ResNet50, SqueezeNet, VGG16, and our proposed framework. The evaluation metrics are color-coded: PPV is shown in yellow, NPV in blue, F1-score in red, and *A* in green. From the graph, it is clear that the proposed framework achieves competitive performance across all metrics. Notably, the proposed framework demonstrates a high accuracy score, comparable to the best-performing models. However, its F1-score, which balances precision and recall, stands out as the highest among all models.

Table 11. Weighted values computation for the predicted metrics using the optimized weight distribution algorithm (preprocessed radiography dataset).

Method	Sp (%)	Rc/Sn (%)	Pr/PPV (%)	NPV (%)	AUC (ROC)	AUC (PR)	F1-score (%)	A (%)
--------	--------	-----------	------------	---------	-----------	----------	--------------	-------

ResNet50	19.36	18.92	18.87	19.36	0.20	0.19	18.91	18.85
ResNet18	19.13	18.58	18.55	19.14	0.20	0.19	18.57	18.58
AlexNet	18.95	17.97	18.20	18.98	0.19	0.19	18.09	18.13
SqueezeNet	19.47	19.21	19.27	19.50	0.20	0.20	19.14	19.10
VGG16	18.79	17.28	18.11	18.95	0.20	0.19	17.65	17.91
Optimized Results	95.70	91.95	93.00	95.93	0.98	0.97	92.36	92.57

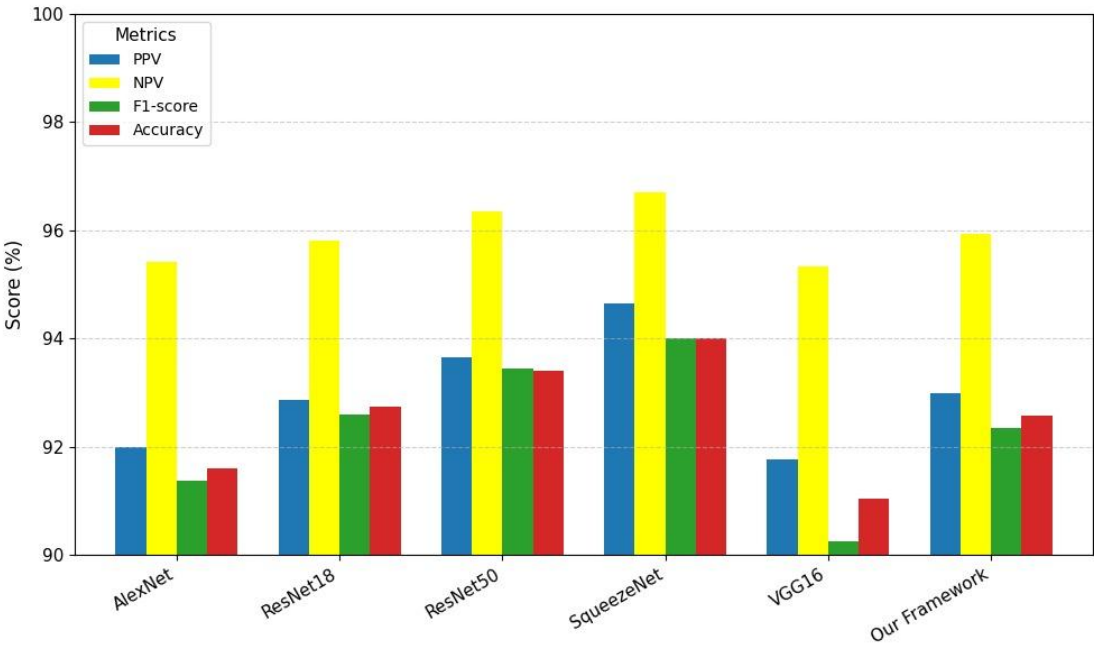


Figure 25. Graphical representation of model evaluation scores. The chart compares the A, PPV, NPV, and F1-score of different models, with the proposed framework.

3.7. Comparison of the Proposed Study with Other Techniques

A comparison of different cohorts for C19-based classification is presented in Table 12. For a fair deduction of the standpoint of the proposed methodology, all the study selections are based on the same dataset or some of its derivatives. Each cohort, of course, is utilizing a different methodology and data split to achieve classification across various classes, including C19, LO, N, and VP. We have used the first three classes to make the task challenging. The last class achieves outstanding results, specifically, due to its discriminative characteristics, which can otherwise lead to negatively biased performance in other classes due to feature dominance.

Samir et al. [63] employed TL with architectures like AlexNet, GoogLeNet, and ResNet-18, combined with a custom CNN. The data is split in an 80/20 ratio, and it classifies four classes, achieving an accuracy of 94.10%. However, this model faces limitations such as bias due to the highly discriminative features of VP. In another study, Wang et al. [64] used MobileNetV3 along with a Dense Block and 5-fold CV. This model classifies three classes, C19, N, and VP, reaching an overall accuracy of 98.71%. Like the previous model, it is limited by VP class-bias from the dataset and cross-testing using just one dataset. Singh et al. [65] used a U-Net architecture for lung segmentation and combined it with a Convolution-Capsule (Conv-Cap) Network. The data is divided into 70% for training, 15% for validation, and 15% for testing. It classifies the same three classes as given, illustrated in [64], but the accuracy drops to 88%. This model also suffers from bias, particularly towards the highly discriminative features of VP ($A = 93\%$).

Khan et al. [66] combined features from TL models with a hybrid whale-elephant herding selection scheme and an extreme learning machine (ELM). The dataset is split into two partitions,

namely training and testing, each of 50%, with a 10-fold CV applied on each. This model classifies four classes, achieving a remarkable accuracy of 99.1%. While its performance is impressive, it is still limited by the fact that the second partition for testing will be used for deriving the results. The training partition carrying 50% of the dataset would not be involved in deriving the results. Furthermore, the use of only one dataset for cross-testing introduces bias due to VP features.

Our proposed framework involves an ensemble of transfer learning with 5-fold cross-validation, classifying three classes, C19, N, and LO, and achieving 92.57% accuracy. The results are robust against model learning bias effects by applying an optimized weight distribution algorithm. Overall, the proposed framework shows competitive performance, though it does not surpass the highest accuracy achieved by [66]. The analysis highlights the common challenge of bias towards VP features and the limitations related to using a single dataset for cross-testing, which affects the generalization ability of the models.

Table 12. Comparison of different models for classifying C19 (using the same dataset).

Reference	Methodology	Data Split	Classes	A	Limitation(s)
[63]	- TL (AlexNet/GoogLeNet/ ResNet-18) - Custom CNN;	80/20 split for both scenarios	single4 (C19, N, VP)	94.10% 89.13%	- Cross testing (only one dataset used) - CV is absent, due to which the generalization check needs further study; - Biasing due to the highly discriminative features of VP
[64]	MobileNetV3 + Dense Block	5-Fold CV	3 (C19, N, VP)	98.71% overall.	- Cross testing (only one dataset used) - Biasing due to the highly discriminative features of VP
[65]	U-Net segmentation, Convolution-Capsule Network	lung 70% training, 15% validation, and 15% testing	3 (C19, N, VP)	88% overall (C19 86%, VP 93%, and N 85%)	- Cross testing (only one dataset used) - Biasing due to the highly discriminative features of VP
[66]	Features from TL model, hybrid whale-elephant selection scheme, Extreme learning machine	Two partitions of 50% each, and 10-Fold CV on each of the partition	4 (C19, N, LO, VP)	99.1%	Partitioning is biased
Proposed framework	Ensemble of Transfer Learning	5-Fold CV	3(C19, N, LO)	92.57%	- Nano-particles as biomarkers - Explainable AI

3.8. Main Findings

The article proposes a DL-based TL method for COVID-19 detection from chest radiographs and CT scans using a weight-optimized distribution algorithm, ensuring that each model’s metric influence is proportional to its performance. It uses an extensive, varied dataset to make the task more challenging, where the strong classification ability of Class VP may negatively affect the performance of Classes C19, LO and N. The discriminative nature of the VP class feature, as illustrated in Figure 26, is clearly observed in the 4-class t-SNE plot using the SqueezeNet pretrained model with 25 epochs.

The study addresses imaging feature variations across five DL models, improving differentiation between C19 manifestations in young and elderly patients. It also offers a non-invasive, cost-effective AI-based diagnostic tool, overcoming the limitations of traditional RT-PCR tests, such as lower sensitivity and longer turnaround times. The proposed framework supports radiologists with reliable second opinions and serves as an educational tool for medical graduates.

3.9. Limitations and Future Recommendations

While this study demonstrates promising results with DL architectures, and those focusing on C19 classification using chest radiographs have made significant progress. However, there are still limitations and areas that need improvement with respect to our study. The following are some key limitations and future recommendations.

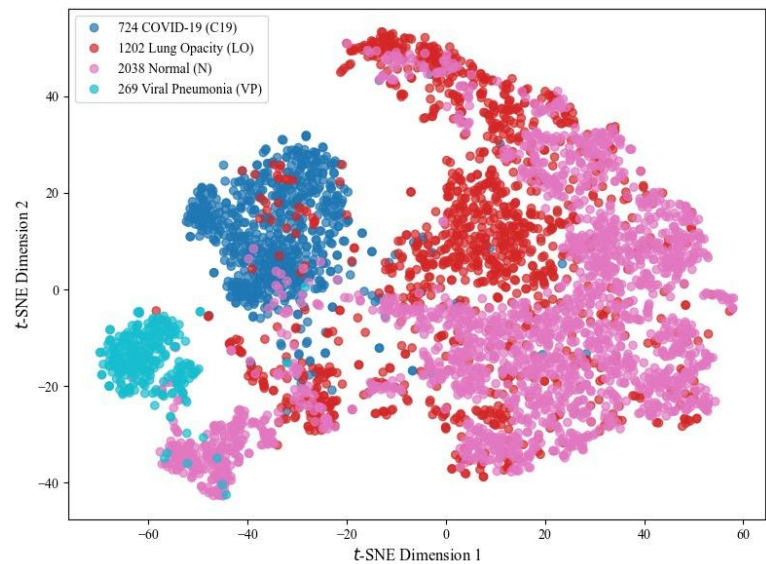


Figure 26. t-SNE plot showing the feature distribution of C19 (724 instances in blue), LO (1202 instances in red), N (2038 instances in pink), and VP (269 instances in cyan) classes run during the testing phase (SqueezeNet, 25 epochs, 5 Fold CV).

One key limitation of this study is the absence of grid computing infrastructure, which could have enhanced the scalability and efficiency of training DL models on large datasets [67]. Without this capability, the training process relied on conventional hardware, potentially limiting the exploration of larger networks or ensemble methods that could improve classification performance. Additionally, the dataset used in this study was neither multi-institutional nor multi-parametric, which may affect the generalizability of the findings. A single-source dataset carries the risk of bias, as it may not fully represent diverse demographic variations, imaging protocols, or disease manifestations seen across different hospitals and populations [68]. Incorporating multi-institutional data with clinical, laboratory, or genomic parameters could have further strengthened the model’s robustness and cross-clinical performance. These limitations highlight opportunities for future work to integrate distributed computing and diverse datasets for more reliable and scalable AI-driven C19 diagnosis.

To improve and expand this work in future, one promising direction is the adoption of explainable artificial intelligence (XAI), which can provide transparent insights into model predictions, highlighting key radiographic features that contribute to C19 diagnosis [69]. Another transformative trend involves the convergence of Nano-diamond-based bio-sensing with DL for multimodal disease assessment. Nano-diamonds offer ultra-sensitive, label-free detection of viral biomarkers, enabling early diagnosis when integrated with AI-driven radiographic analysis [70]. Moreover, Radiogenomic approaches can link imaging phenotypes with genomic data, enabling precision medicine in C19 management [71]. By training DL models on multi-modal datasets that incorporate chest X-rays, genetic risk factors, and proteomic biomarkers, AI systems could predict disease severity. Additionally, large deep learning models (ResNet50, ResNet18, VGG16, AlexNet, SqueezeNet), which contain a large number of learnable parameters, these could potentially be reduced by utilizing lightweight architectures [72].

4. Conclusions

In this study, we introduced an ensemble approach of five advanced transfer learning models designed to improve the diagnosis of COVID-19 from chest radiographs. With the limitations of the RT-PCR test, especially in clinical situations, our proposed solution offers a reliable alternative for accurate classification. The weighted optimization strategy ensured that models with superior performance were given more influence, resulting in enhanced prediction accuracy. The SqueezeNet model achieved a high accuracy of 94.01%, while the ensemble approach demonstrated 92.57% accuracy and an F1-score of 92.36%. These results highlight the effectiveness of the model in addressing biases and providing reliable diagnostic support. This framework not only streamlines radiology workflows but also aids in advancing artificial intelligence in healthcare, offering a valuable tool for medical professionals and supporting the development of AI expertise among medical graduates.

References

1. Binny, R.N., et al., *Sensitivity of reverse transcription polymerase chain reaction tests for severe acute respiratory syndrome coronavirus 2 through time*. The Journal of infectious diseases, 2023. **227**(1): p. 9-17.
2. Koh, H.K., A.C. Geller, and T.J. VanderWeele, *Deaths from COVID-19*. Jama, 2021. **325**(2): p. 133-134.
3. Eccleston, C., et al., *Managing patients with chronic pain during the COVID-19 outbreak: considerations for the rapid introduction of remotely supported (eHealth) pain management services*. Pain, 2020. **161**(5): p. 889-893.
4. Yuan, Y., et al., *The development of COVID-19 treatment*. Frontiers in immunology, 2023. **14**: p. 1125246.
5. Christie, A., *Decreases in COVID-19 cases, emergency department visits, hospital admissions, and deaths among older adults following the introduction of COVID-19 vaccine—United States, September 6, 2020–May 1, 2021*. MMWR. Morbidity and mortality weekly report, 2021. **70**.
6. Shereen, M.A., et al., *COVID-19 infection: Emergence, transmission, and characteristics of human coronaviruses*. Journal of advanced research, 2020. **24**: p. 91-98.
7. Dong, D., et al., *The role of imaging in the detection and management of COVID-19: a review*. IEEE reviews in biomedical engineering, 2020. **14**: p. 16-29.
8. Kolhar, M., A.M. Al Rajeh, and R.N.A. Kazi, *Augmenting Radiological Diagnostics with AI for Tuberculosis and COVID-19 Disease Detection: Deep Learning Detection of Chest Radiographs*. Diagnostics, 2024. **14**(13): p. 1334.
9. Jouibari, Z.E., H.N. Moakhkhar, and Y. Baleghi, *Emergency covid-19 detection from chest x-rays using deep neural networks and ensemble learning*. Multimedia Tools and Applications, 2024. **83**(17): p. 52141-52169.
10. Srinivas, K., et al., *COVID-19 prediction based on hybrid Inception V3 with VGG16 using chest X-ray images*. Multimedia Tools and Applications, 2024. **83**(12): p. 36665-36682.
11. Kumar, M., S.D. SL, and B. Prashanth, *CXNet-A Novel approach for COVID-19 detection and Classification using Chest X-Ray image*. Procedia Computer Science, 2024. **235**: p. 2486-2497.
12. Fernández-Miranda, P.M., et al., *A retrospective study of deep learning generalization across two centers and multiple models of X-ray devices using COVID-19 chest-X rays*. Scientific Reports, 2024. **14**(1): p. 14657.

13. Brunese, L., et al., *Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays*. Computer Methods and Programs in Biomedicine, 2020. **196**: p. 105608.
14. Nayak, S.R., et al., *Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: A comprehensive study*. Biomedical Signal Processing and Control, 2021. **64**: p. 102365.
15. Zhang, J., et al., *Viral pneumonia screening on chest X-rays using confidence-aware anomaly detection*. IEEE transactions on medical imaging, 2020. **40**(3): p. 879-890.
16. Arias-Londono, J.D., et al., *Artificial intelligence applied to chest X-ray images for the automatic detection of COVID-19. A thoughtful evaluation approach*. Ieee Access, 2020. **8**: p. 226811-226827.
17. Vaid, S., R. Kalantar, and M. Bhandari, *Deep learning COVID-19 detection bias: accuracy through artificial intelligence*. International Orthopaedics, 2020. **44**: p. 1539-1542.
18. Kassania, S.H., et al., *Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: a machine learning based approach*. Biocybernetics and Biomedical Engineering, 2021. **41**(3): p. 867-879.
19. Hussain, E., et al., *CoroDet: A deep learning based classification for COVID-19 detection using chest X-ray images*. Chaos, Solitons & Fractals, 2021. **142**: p. 110495.
20. Ismael, A.M. and A. Şengür, *Deep learning approaches for COVID-19 detection based on chest X-ray images*. Expert Systems with Applications, 2021. **164**: p. 114054.
21. Xu, X., et al., *A deep learning system to screen novel coronavirus disease 2019 pneumonia*. Engineering, 2020. **6**(10): p. 1122-1129.
22. Ardakani, A.A., et al., *Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks*. Computers in biology and medicine, 2020. **121**: p. 103795.
23. Li, L., et al., *Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT*. Radiology, 2020: p. 200905.
24. Çiçek, Ö., et al. *3D U-Net: learning dense volumetric segmentation from sparse annotation*. in *International conference on medical image computing and computer-assisted intervention*. 2016. Springer.
25. Hu, R., et al., *Automated diagnosis of covid-19 using deep learning and data augmentation on chest ct*. Medrxiv, 2020: p. 2020.04. 24.20078998.
26. Wang, L., Z.Q. Lin, and A. Wong, *Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images*. Scientific reports, 2020. **10**(1): p. 19549.
27. Butt, C., et al., *Deep learning system to screen coronavirus disease 2019 pneumonia*. Appl Intell, 2020.
28. Bondugula, R.K., S.K. Udgata, and N.S. Bommi, *A novel weighted consensus machine learning model for covid-19 infection classification using CT scan images*. Arabian Journal for Science and Engineering, 2023. **48**(8): p. 11039-11050.
29. Sinra, A. and H. Angriani, *Automated Classification of COVID-19 Chest X-ray Images Using Ensemble Machine Learning Methods*. Indonesian Journal of Data and Science, 2024. **5**(1): p. 45-53.
30. Hussein, A.M., et al., *Auto-detection of the coronavirus disease by using deep convolutional neural networks and X-ray photographs*. Scientific reports, 2024. **14**(1): p. 534.
31. Chen, T., et al., *A vision transformer machine learning model for COVID-19 diagnosis using chest X-ray images*. Healthcare Analytics, 2024. **5**: p. 100332.
32. Talukder, M.A., et al., *Empowering covid-19 detection: Optimizing performance through fine-tuned efficientnet deep learning architecture*. Computers in Biology and Medicine, 2024. **168**: p. 107789.
33. Bhatele, K.R., et al., *Covid-19 detection: A systematic review of machine and deep learning-based approaches utilizing chest x-rays and ct scans*. Cognitive Computation, 2024. **16**(4): p. 1889-1926.
34. Nair, S.S.K., et al., *CovMediScanX: A medical imaging solution for COVID-19 diagnosis from chest X-ray images*. Journal of Medical Imaging and Radiation Sciences, 2024. **55**(2): p. 272-280.
35. Yadlapalli, P. and D. Bhavana, *Application of Fuzzy Deep Neural Networks for Covid 19 diagnosis through chest Radiographs*. F1000Research, 2023. **12**(60): p. 60.
36. Dokumacı, H.Ö., *DERİN ÖĞRENME TABANLI MODELLERLE AKCİĞER X-RAY GÖRÜNTÜLERİNDEN COVID-19 TESPİTİ*. Kahramanmaraş Sütçü İmam Üniversitesi Mühendislik Bilimleri Dergisi, 2024. **27**(2): p. 481-487.

37. Rahman, T., et al., *Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images*. Computers in biology and medicine, 2021. **132**: p. 104319.
38. Chowdhury, M.E., et al., *Can AI help in screening viral and COVID-19 pneumonia?* Ieee Access, 2020. **8**: p. 132665-132676.
39. Gonzalez, R.C., *Digital image processing*. 2009: Pearson education india.
40. Mallat, S., *A wavelet tour of signal processing*. 1999: Elsevier.
41. Tomasi, C. and R. Manduchi, *Bilateral Filtering for Gray and Color Images*, in *Proceedings of the Sixth International Conference on Computer Vision*. 1998, IEEE Computer Society. p. 839.
42. Nitish, S., *Dropout: a simple way to prevent neural networks from overfitting*. J. Mach. Learn. Res., 2014. **15**: p. 1.
43. Nwankpa, C., et al., *Activation functions: Comparison of trends in practice and research for deep learning*. arXiv preprint arXiv:1811.03378, 2018.
44. Agarap, A.F., *Deep learning using rectified linear units (relu)*. arXiv preprint arXiv:1803.08375, 2018.
45. Jang, E., S. Gu, and B. Poole, *Categorical reparameterization with gumbel-softmax*. arXiv preprint arXiv:1611.01144, 2016.
46. Smith, L.N., *A disciplined approach to neural network hyper-parameters: Part 1--learning rate, batch size, momentum, and weight decay*. arXiv preprint arXiv:1803.09820, 2018.
47. Akbar, S., et al. *Transitioning between convolutional and fully connected layers in neural networks*. in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*. 2017. Springer.
48. Yamashita, R., et al., *Convolutional neural networks: an overview and application in radiology*. Insights into imaging, 2018. **9**: p. 611-629.
49. Santurkar, S., et al., *How does batch normalization help optimization?* Advances in neural information processing systems, 2018. **31**.
50. Pérez, M., *An Investigation of ADAM: A Stochastic Optimization Method*, in *Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022*. 2022.
51. Klingler, N., *AlexNet: A Revolutionary Deep Learning Architecture*. viso. ai. [https://viso. ai/deep-learning/alexnet/](https://viso.ai/deep-learning/alexnet/). Accessed, 2023. **13**.
52. Flach, P. and M. Kull, *Precision-recall-gain curves: PR analysis done right*. Advances in neural information processing systems, 2015. **28**.
53. He, K., et al. *Deep residual learning for image recognition*. in *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
54. Iandola, F.N., et al., *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size*. arXiv preprint arXiv:1602.07360, 2016.
55. Simonyan, K. and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556, 2014.
56. Rainio, O., J. Teuvo, and R. Klén, *Evaluation metrics and statistical tests for machine learning*. Scientific Reports, 2024. **14**(1): p. 6086.
57. Erickson, B.J. and F. Kitamura, *Magician's corner: 9. Performance metrics for machine learning models*. 2021, Radiological Society of North America. p. e200126.
58. Swift, A., R. Heale, and A. Twycross, *What are sensitivity and specificity?* Evidence-Based Nursing, 2020. **23**(1): p. 2-4.
59. Altman, D.G. and J.M. Bland, *Diagnostic tests. 1: Sensitivity and specificity*. BMJ: British Medical Journal, 1994. **308**(6943): p. 1552.
60. Miao, J. and W. Zhu, *Precision-recall curve (PRC) classification trees*. Evolutionary intelligence, 2022. **15**(3): p. 1545-1569.
61. Derczynski, L. *Complementarity, F-score, and NLP Evaluation*. in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. 2016.
62. Qureshi, S.A., et al., *SAlexNet: Superimposed AlexNet using residual attention mechanism for accurate and efficient automatic primary brain tumor detection and classification*. Results in Engineering, 2025. **25**: p. 104025.

63. Samir, B., et al., *Deep learning for classification of chest X-ray images (Covid 19)*. arXiv preprint arXiv:2301.02468, 2023.
64. Wang, S., J. Ren, and X. Guo, *A high-accuracy lightweight network model for X-ray image diagnosis: A case study of COVID detection*. Plos one, 2024. **19**(6): p. e0303049.
65. Singh, T., et al., *COVID-19 severity detection using chest X-ray segmentation and deep learning*. Scientific Reports, 2024. **14**(1): p. 19846.
66. Khan, M.A., et al., *COVID-19 classification from chest X-ray images: a framework of deep explainable artificial intelligence*. Computational Intelligence and Neuroscience, 2022. **2022**(1): p. 4254631.
67. Baker, M., R. Buyya, and D. Laforenza, *Grids and Grid technologies for wide-area distributed computing*. Software: Practice and Experience, 2002. **32**(15): p. 1437-1466.
68. Rouzrokh, P., et al., *Mitigating bias in radiology machine learning: 1. Data handling*. Radiology: Artificial Intelligence, 2022. **4**(5): p. e210290.
69. Pennisi, M., et al., *An explainable AI system for automated COVID-19 assessment and lesion categorization from CT-scans*. Artificial intelligence in medicine, 2021. **118**: p. 102114.
70. Qureshi, S.A., et al., *Recent development of fluorescent nanodiamonds for optical biosensing and disease diagnosis*. Biosensors, 2022. **12**(12): p. 1181.
71. Qureshi, S.A., et al., *Radiogenomic classification for MGMT promoter methylation status using multi-omics fused feature space for least invasive diagnosis through mpMRI scans*. Scientific reports, 2023. **13**(1): p. 3291.
72. Qureshi, S.A., et al., *Intelligent ultra-light deep learning model for multi-class brain tumor detection*. Applied Sciences, 2022. **12**(8): p. 3715.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.