JCBI

Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606

Volume 07 Issue 01
2024

Research Article
https://doi.org/10.56979/701/2024

A Genetic Algorithm for Targeted Regression Testing: Balancing Coverage and Change Focus

Fahad Nazir[1], Tehreem Masood[1, 2], Hafiz Muhammad Tayyab Khushi[1,2], Shamim Akhter[3], and Iftikhar Naseer[1,2*]

[1]Faculty of Computer Science & Information Technology, Superior University, Lahore, 54000, Pakistan.
[2]Department of Software Engineering, Superior University, Lahore, 54000, Pakistan.
[3]School of Information Management, Minhaj University, Lahore, 54000, Pakistan.
[*]Corresponding Author: Iftikhar Naseer. Email: Iftikhar.naseer@superior.edu.pk

Received: January 21, 2024 **Accepted:** May 02, 2024 **Published:** June 01, 2024

Abstract: The selection of regression test cases is used to choose a subset of test suits that are used to exercise the altered program to ensure that the modified part has no unintended consequences on the unmodified part of the program. In previous works, the single objective is used for the selection of test cases. In this thesis, we preserved test case selection as a multi-objective optimization problem. We select code coverage and code change information for test case selection. There are many different methods for test case assortment (i.e., firewall, textual differing g, test-tube, etc.). We used a genetic procedure for multi-objective test case assortment. Our proposed technique first collects the related information such as the size of the system under test, the size of the test suit, and modification between the original and modified versions of the system. Then collect and analyze the code coverage and code change information, also collect user requirements for test case selection. Finally, select a subset of test cases based on cost, fault uncovering, code coverage, and code alteration information for test case selection. The proposed system is for a moderate-level desktop application. Three datasets (Triangle, Tree data structure, and Jodatime) are used for the experiment of the proposed system. For the evaluation of the proposed test selection technique, we used precision and recall evaluation matrices. Our experiment study demonstrates that our proposed technique selects almost 75% of related test cases.

Keywords: Regression Testing; Check Case Selection; Genetic Algorithm; Test Case Selection Technique.

1. Introduction

Software testing is vital to test the software that satisfies user specifications and requirements. Software testing is a process to check whether the result which is produced by software, matches with expected results or not and also ensures that the developed system is fault free. Software is modified to fix the bug, a new fea-ture is added according to the new customer requirements, or some existing features are updated. Changes that apply in the new version according to customer requirements or fixing the bug require confirmation that all the changes that are made in the new version have not affected the functionality of the old version for this confir-mation regression testing is used. Regression testing is carried out at several stages of system development, in-cluding unit testing, integration testing, system testing, and maintenance. Retesting all of the regression test cases will need extra time, money, and resources. Due to time and resource constraints, comprehensive test suite execution in regression testing is not always viable. To lower the cost of regression testing, a selection of test cases from a test suite that can find flaws or faults must be chosen. The capacity to decrease maintenance expenses through accurate regression test case selection. Regression Testing processes include test suit mini-mization (TSM), test case selection (TCS), and test case prioritization (TCP) [1]. This study is based on the re-gression testing test case selection approach.

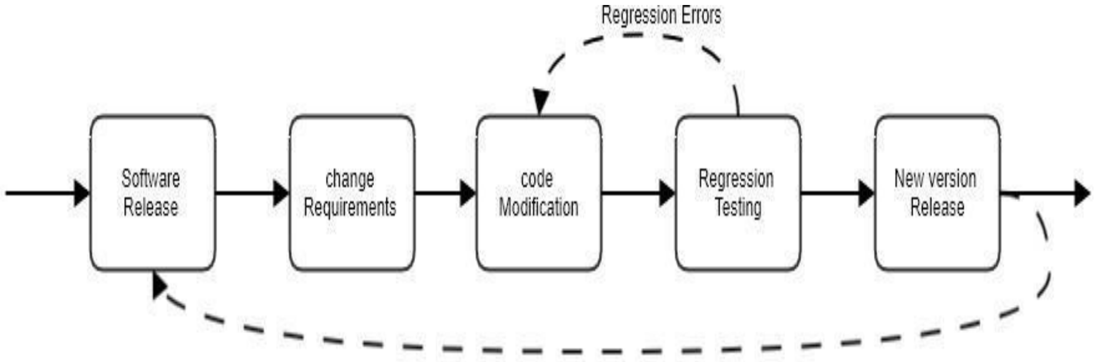


Figure 1. .Maintainance Process Model[1]

Test case selection (TCS) approaches choose a selected group of test scenarios from the test suite. It chooses test cases relating to the new software component. TSM, TCP, and TCS vary in that TSM discovers unemployed or excess test cases and eliminates those from the testing suite to downsize the test suite, but TCS does not delete test cases; rather, it picks test cases connected with the updated component of the newly released version of the product. TCP techniques prioritize test cases based on factors such as cumulative fault exposure and service, among others. Test cases are prioritized to detect software faults as soon as feasible.

1.1 Regression Testing

Regression testing confirms that new faults will not present in enlarged code or requirement variations [2]. Regression testing is required in the following circumstances.

- New features are add-in software
- New functions are added in the software according to customer requirements
- Fixing bugs that are found in the previous version of the software
- Remove unwanted features.

There are some advantages of regression testing:

- Regression testing improves the quality of the product by tests.
- Notify us if any side effect occurred due to modification in the code.
- Guarantee that the bugs discovered before don't rise.

Regression testing can be done manually and many tools are available to automate regression testing. Regression test case selection was introduced by Fisher for the preservation of software, the test cases are chosen for regression testing, rerun these test cases, and validate the changes that are made in the new version, to ensure the software is error-free. Rerun selected test cases are less expensive in terms of cost of execution, maintenance, and development in its place of rerunning all the test cases in the test suite. Microsoft defines a test case as a Test case is a unit of partial testing derived from the exploration with the selected traversal criteria. The test suite is a grouping of test cases, it can be enhanced according to the expansion of the software. Regression Test case Selection (RTS) is stabbed to upsurge the efficiency of testing grounded on measurement abilities such as coverage, cost, and fault detection [2].

1.2. Two activities are performed in an RTS

- Elect the modified part of the program
- Select the test cases that test these modified parts of the program.

The purpose of RTS is to reduce the cost of testing by lowering the cost of execution. When the cost of the RTS technique is compared to the retest-all strategy, this approach is unfeasible owing to its efficacy and test cost. The alternative strategy of Retest-all is a random selection of test cases; however, the disadvantage of this approach is that all flaws are not identified owing to randomly picked test cases. The RTS technique is used to eliminate the drawbacks of both the retest-all approach and the random select test cases approach by selecting the most relevant test cases based on adequacy criteria. Some common adequacy criteria used in regression test case selection technique are coverage, fault, and cost-based, as well as different groupings of all these adequacy criteria.

Regression test case selection contributes to overall software quality, software system maintenance, software updates, product dependability, and product deployment activities, among other things.

RTS and RTM problems are similar, in both problems, taking a subset of test cases from the test suite. The main difference is that the Regression test case reduction is based on data like coverage from a single version of the product. Select regression test cases because their execution is important to the differences between the prior and current versions of the program.

The selection approach varies depending on how a given technique is defined, and it searches and identifies changes in the program that will be tested. CFG cluster identification, textual difference, data flow analysis, module level firewall, the control flow analysis-based technique, use case diagram-based technique, path analysis, UML architecture-based technique, and many other approaches have been proposed.

1.3. Regression Testing Strategies

There are two main regression testing strategies on how to reprocess test cases for the altered program. Retest all and selective retest are regression testing strategies. The simple and upfront method of Regression testing is the Retest-all approach. In the retest-all approach, all the test cases are re-run to test the modified code. This approach is safe because all the modified code is excised. Retesting all approaches is expensive and time-consuming. The alternative to this strategy is a selective retest.

Selective retest, Select those test cases after the whole test set that are connected to the altered program. This technique reduces the time to test the altered program because the only modified program is exercised by a subdivision of selective test cases. They identify two issues in the selective retest approach one is in what way to select a test case after the initial test suite and the other is how to recognize where additional test cases may be required.

1.4. Categories of Regression Testing Technique

Three regression testing strategies are used to lower the cost and recover the competence of regression testing. These regression difficult strategies include test case selection, ordering, and minimization [5].

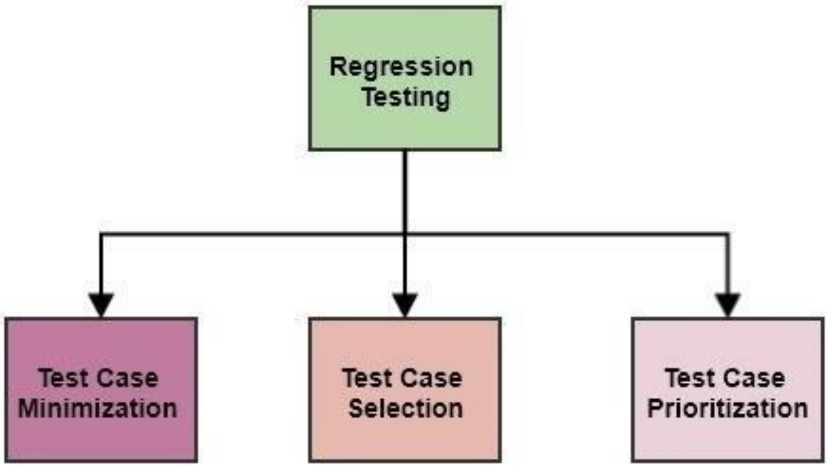


Figure 2. Techniques of Regression Testing

Test case selection (TCS) approaches choose a selected group of test scenarios from the test suite. It chooses test cases relating to the new software component. TSM, TCP, and TCS vary in that TSM discovers unemployed or excess test cases and eliminates those from the testing suite to downsize the test suite, but TCS does not delete test cases; rather, it picks test cases connected with the updated component of the newly released version of the product. TCP techniques prioritize test cases based on factors such as cumulative fault exposure and service, among others. Test cases are prioritized to detect software faults as soon as feasible.

It is critical to diagnose bacterial or viral pneumonia as soon as possible and to start therapy as soon as possible ,can greatly diminish the probability of a patient's condition worsening and resulting in mortality. [2]. Currently, the most reliable way to identify pneumonia is thought to be a chest X-ray[3]. Pneumonia is not always evident on X-rays, and it is sometimes mistaken for other benign abnormalities or other diseases. Furthermore, specialists may misclassify images of bacterial or viral pneumonia, which could result in patients receiving the incorrect medicine and a deterioration of their condition [4-6]. Significant subjective discrepancies have been observed in the diagnoses of pneumonia made by radiologists. There is a chronic lack of qualified radiologists in low-resource countries (LRC), particularly in rural regions. Computer-aided detection (CAD) solutions are desperately needed to help radiologists meet this issue,identify different types of pneumonia, as they rely on chest X-ray pictures for this purpose.

Artificial Intelligence (AI)-based solutions are currently being used for several biomedical issues (e.g., brain tumour identification, detecting breast cancer. [7-10]. One class of deep learning methods is Convolutional Neural Networks (CNNs),that have gained a lot of traction in the scientific community due to their impressive image classification capabilities [11]. Deep Learning: Deep learning methods for machine learning from chest X-rays are becoming more and more popular since they are simple to apply with inexpensive imaging techniques and have a wealth of data to train various models. Among the various research organizations that have utilized deep learning algorithms for pneumonia diagnosis, only one has specifically documented the ability to differentiate between bacterial and viral pneumonia [12], while whereas a number of research teams [1] [12-25] describe how deep learning algorithms can be used to diagnose pneumonia.

In numerous studies, writers have experimented with adjusting the deep layered CNN's settings in order to identify pneumonia. Radiographs of the lungs obtained for the purpose of diagnosing pneumonia may show interstitial or alveolar patterns of widespread opacity. In a laboratory context, the presence of alveolar infiltration—particularly lobar infiltrates—on a chest radiograph is indicative of bacterial infection.[26]. In our research on Pneumonia Recognition in Chest X-rays through Convolutional Neural Networks (CNN), we draw inspiration from a novel deep learning technique proposed in a paper focusing on highly accurate image retrieval, incorporating auto-correlation, gradient computation, and feature fusion. Once more, viral pneumonia may be associated with interstitial infiltrations on radiographs. This work proposes a unique system using blockchain-based federated learning to quickly detect and isolate COVID-19 situations, gathering information from many sources while protecting privacy concerns. The suggested model, which included IELMs and CapsNet, showed better accuracy (98.99%) when it came to COVID-19 patient classification using chest CT images. This paper presents CDC Net, a convolutional neural network model, in the context of detecting numerous chest infections. It achieves an AUC of 0.9953, accuracy of 99.39%, recall of 98.13%, and precision of 99.42%. Its higher performance is validated by comparative comparisons with other CNN-based pre-trained models.These may be the distinguishing characteristics of the machine learning algorithms utilised to differentiate between viral and bacterial pneumonia. Promising outcomes have been reported by certain researchers, Numerous scholars have investigated this subject, including Xianghong et al., Liang et al., as well as Krishnan et al, and Vikash et al.Multiple teams utilized CNN models incorporating Teams used lung field segmentation and rib suppression approaches to meet different categorization issues. In contrast, other teams prioritized the utilization of CNN visualization techniques to Determine the child-specific Region of Interest (ROI) in chest X-rays, enabling the detection of pneumonia and differentiation between viral and bacterial types.

Vikash et al. employed ensembles of pre-trained ImageNet models [27] to identify pneumonia utilising the deep learning framework's idea of transfer learning. In Xianghong et al. a modified version of the VGG16 model was employed. The deep convolutional neural network is employed for the classification of pneumonia, while the fully convolutional network (FCN) is used to find and identify the lung sections.

Wang et al. [13] utilised a deep learning approach on an X-ray dataset of 32,717 patients, with encouraging results.

The authors of [14] continued to use CNN in conjunction with data augmentation to improve performance through training on a limited number of images. Using an ensemble of several networks, Rajpurkar et al. identified pneumonia as one of fourteen illnesses using 121 layer CNN developed using chest X-ray data. To solve the gradient vanishing problem, The 3D Deep Convolutional Neural Network was utilised by Jung et al. with dense connections and shortcuts. Furthermore, The Region-based CNN is a deep neural network that was employed by Jaiswal et al. [17]and Siraz et al. to categorise lung images. By utilizing the RetinaNet ensemble and implementing picture enhancing techniques, the precise location of the pneumonia may be accurately identified. In, fourteen thoracic illnesses were accurately identified by the application of feature extraction techniques and a pretrained DenseNet-121.

Using AlexNet and GoogLeNet, Sundaram et al. used data augmentation strategies to get an AUC value between 0.94 and 0.95. The authors achieved superior performance compared to their competitors by making modifications to a two-network ensemble architecture, as demonstrated by the remarkable 0.99 area under the curve (AUC) value.

When employing deep learning algorithms to classify distinguishing between bacterial and cases of viral pneumonia as well as normal instances. utilizing X-rays to diagnose pneumonia pictures, the maximum accuracy recorded in the aforementioned literatures was 93.6% and 96.84%, respectively. As a result, there is a lot of opportunity to improve the outcome, especially in identifying bacterial and viral pneumonia, by combining multiple outperforming algorithms into an ensemble model, using new deep learning algorithms or tweaking the existing ones that are producing better results. The proposed paper presents a transfer learning approach and evaluates the capabilities of SqueezeNet, AlexNet, ResNet18, and DenseNet201, four distinct pre-trained network designs. This study's most significant contribution is a transfer-learning strategy based on convolutional neural networks (CNNs), which uses multiple pre-trained algorithms can more precisely detect and categorise viral and bacterial pneumonia than earlier studies. Furthermore, the details of the methodology used in the paper are provided in the publication, which any research group can use to benefit from this work. Furthermore, radiographic results are still not very good at pointing to the source of pneumonia. The reason for conducting this research was to better distinguish between bacterial and viral pneumonia by using machine learning to diagnose pneumonia using radiograph analysis in the first place. The rest of the paper is divided into the following sections: Section 3 outlines the study's methodology and includes information on the dataset's specifics as well as the pre-processing procedures that were taken to get the data ready for testing and training. Section 2 lists the several pre-trained networks that were used for the investigation. The findings of the classification algorithms are given in Section 4, where they are also compared with those of a few other recent studies and discussed. Lastly, Section 5 presents the conclusion.

2. Overview of the History of Algorithms for Deep Machine Learning

2.1. CNNs (Convolutional Neural Networks)

The enhanced image categorization capabilities of Convolutional Neural Networks (CNNs) are a significant factor contributing to their rapid surge in popularity. The network's convolutional layers and filters aid in the more accurate extraction of spatial and temporal aspects of images. Implementing a method to evenly distribute the weight between the layers aids in minimizing computing demands.

With two limitations on their architecture, CNNs are essentially feedforward artificial neural networks (ANNs): first, to preserve spatial structure, Furthermore, to reduce the overall count of model parameters, the neurons' weights inside the same filter are shared. Furthermore, neurons within the same filter exhibit connections exclusively with adjacent patches of the image. The network's ability to classify is enabled by a fully connected layer, dimensionality reduction is achieved through a max-pooling (subsampling) layer, and feature learning is performed by a convolutional layer are the three basic components of a CNN. An overview of the architecture of CNN is shown in Figure 3.

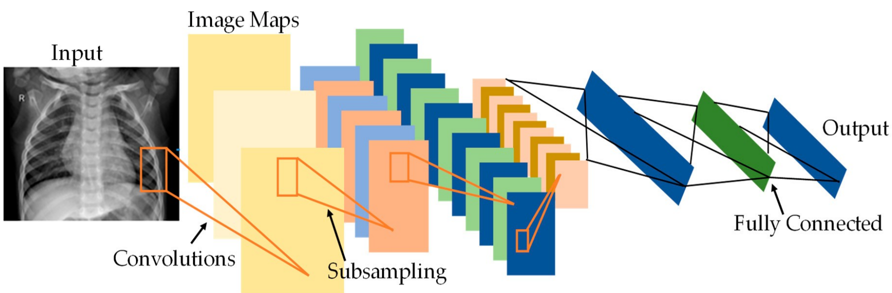


Figure 3. CNN Architecture.

2.2. Transfer Learning in Deep CNN

In general, Larger datasets yield superior performance for CNNs than smaller ones. When using CNN in applications , transfer learning can be useful when dealing with tiny datasets.

4 illustrates the conceptualizing knowledge of transfer learning by demonstrating how a trained model from a larger dataset—like ImageNet— It is possible to utilize with a limited dataset. Recently, transfer learning—applying knowledge from one job to another—has demonstrated promise in real-world industries including manufacturing, healthcare, and baggage screening.This eliminates the need for a big

dataset and shortens the training period, both of which are necessary for the deep learning algorithm during its construction from the ground up.

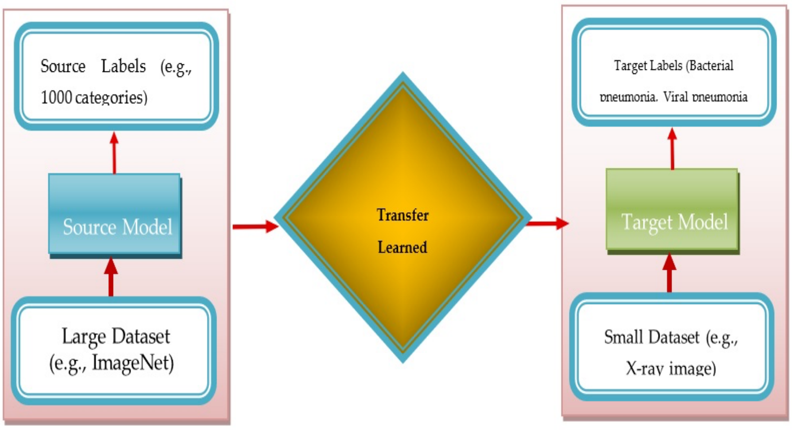


Figure 4. TL(transfer learning) concepts.

2.3. Pre-Trained CNNS

Four well-known convolutional neural networks AlexNet[11], DenseNet[28], ResNet[29], and SqueezeNet[30] were employed in this study to do deep learning. The following was a a succinct synopsis of these trained networks

2.3.1. AlexNet

To classify over a thousand distinct types, AlexNet employs deep layers with sixty million parameters and six hundred thousand neurons. There are a total of six levels in the network: two FLCs, Five CLs, One Softmax layer and three pooling layers make up the architecture [11]. The dimensions of the supplied picture should be 227 × 227 × 3. in order for AlexNet to function. The first CL uses 96 kernels with an 11 × 11 × 3 size and a four pixel stride to alter the input image. The data is sent into the second layer, and Figure 5 summarises the remaining information.

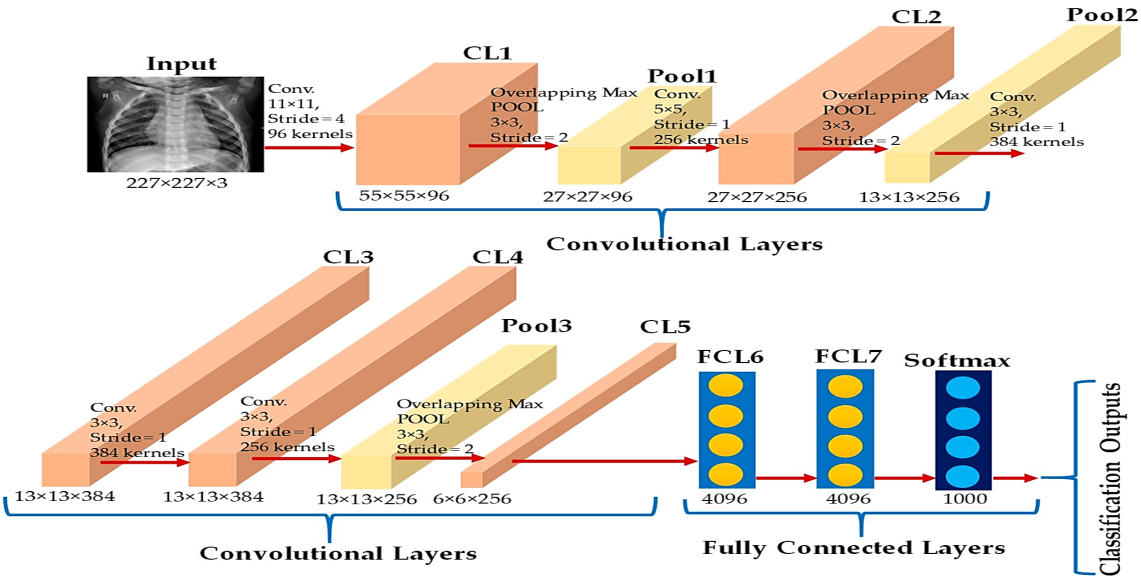


Figure 5. Structure of AlexNet

2.3.2. ResNet18

The initial architecture of ResNet, an abbreviation for Residual Network, which was created to tackle two problems, deterioration and the vanishing gradient, is shown in Figure 6 [29]. Relative learning aims to tackle both of these issues. There are three versions of ResNet available: ResNet18, ResNet50, and ResNet101. Each version differs from the other based on the number of layers. ResNet has demonstrated efficacy in utilizing transfer learning for classifying biological images[30]. This research employed ResNet18 to detect instances of pneumonia. During training, ResNet prioritizes residuals over features, which sets it apart from deep neural networks' layers that usually learn either basic or intricate features.

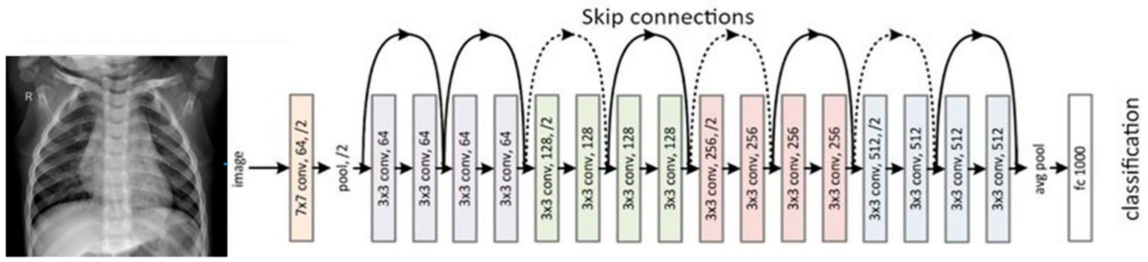


Figure 6. Structure of ResNet18.

2.3.3. DenseNet201

Because it doesn't train redundant feature maps, DenseNet, using fewer parameters than a standard CNN, a condensed version of the Dense Convolutional Network [31]. With just 12 filters, DenseNet produces a negligible amount of new feature maps. DenseNet121, DenseNet169, DenseNet201, and DenseNet264 are the four distinct variations of DenseNet. DenseNet201 was utilised in this paper to diagnose pneumonia [28]. DenseNet (shown in Figure 7) offers easy and quick access to the original input picture for each layer in addition to gradients obtained using the loss function. DenseNet is a superior option for picture classification as a result of the considerable reduction in computing cost.

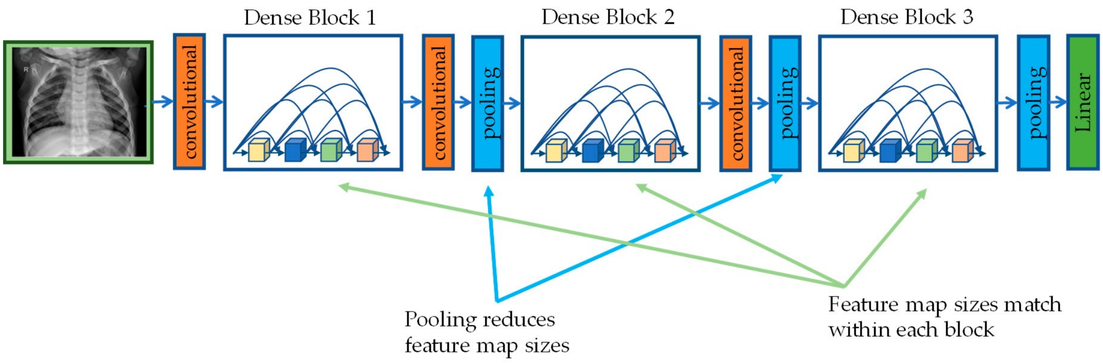


Figure 7. DenseNet201 architecture

2.3.4. SqueezeNet

In addition to SqueezeNet, another convolutional neural network (CNN) was trained utilizing the ImageNet database[28]. SqueezeNet, having undergone training with over a million images, possesses a parameter count that is fifty times smaller than that of AlexNet. The expand layer and squeeze layer together form the module of fire, which is the network's core element. The extend layer receives data from the squeeze layer and applies a combination of 1 × 1 and 3 × 3 convolutional filters. Here, only 1 × 1 convolutional filters are employed[32]. This is illustrated in Figure 8. Here, In order to distinguish between presentations of bacterial and viral illnesses and to diagnose pneumonia, we employ the pre-trained SqueezeNet model.

3. Methodology

3.1. Dataset

5247 chest X-ray pictures in all, ranging in resolution from 400p to 2000p, were extracted from the database of Kaggle Chest X-ray Pneumonia for this study[33]. Table 1 illustrates that of the 5247 chest X-ray images, 1341 are of healthy individuals and 3906 are of those suffering from pneumonia, comprising 2561 images of pneumonia bacteria and 1345 images of pneumonia viruses. Some episodes of pneumonia may involve a combination of viral and bacterial infections. No instances of co-infection between bacteria and viruses were found in the dataset utilized for this study. This dataset was utilized to segregate a training set from a test set..

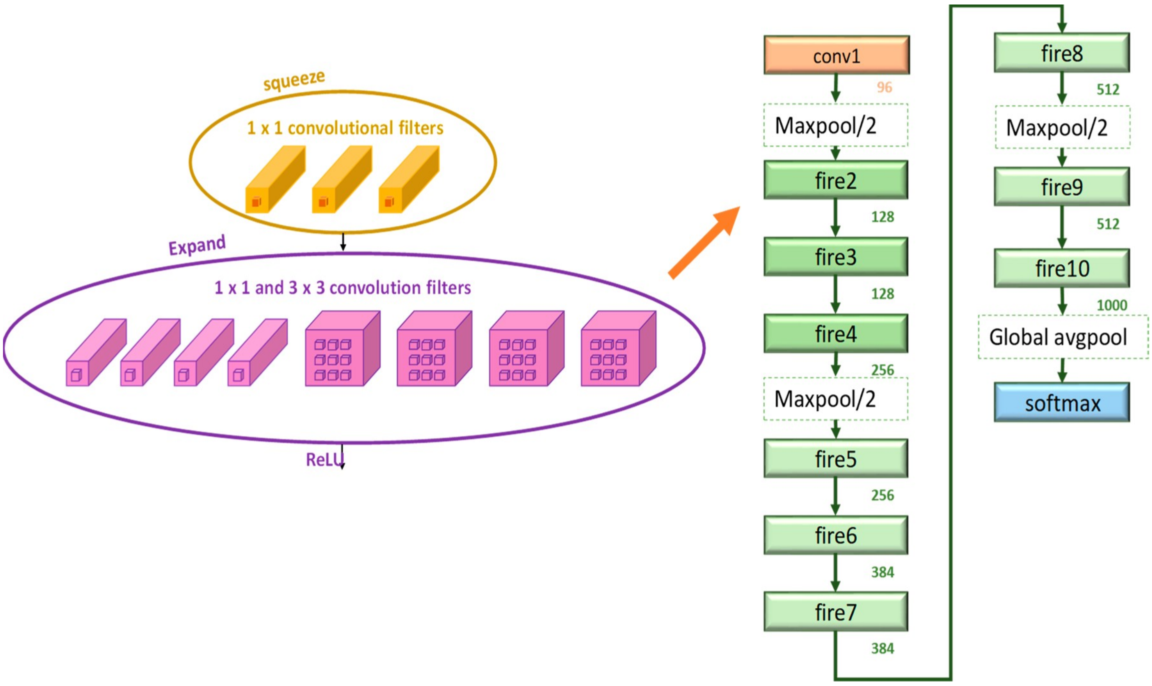


Figure 8. Architecture of SqueezeNet

Table 1. Complete dataset details.

Cases	**Type of Images**	**No of X-ray Images**
Case A	Normal Cases	1341
Case B	Bacterial Pneumonia Cases	2561
Case C	Viral Pneumonia Cases	1345
Total Cases	Total	5247

A variety of test and train images (using augmentation) for various evaluation trials are displayed in table 2 Using the training dataset, four distinct algorithms were trained and subsequently assessed using the test dataset. Two samples of chest X-rays for viral, bacterial, and normal pneumonia are shown in figure 9.

Table 2. Information about the test and training sets.

Train Images	**Test Images**	**Cases**	**Types**
4500	205	**Normal**	Normal and Pneumonia
4500	214	**Pneumonia**	
4500	199	**Normal**	Normal, Bacterial, and Viral Pneumonia
4500	197	**Bacterial Pneumonia**	
4500	201	**Viral Pneumonia**	
4500	197	**Bacterial Pneumonia**	Bacterial and Viral Pneumonia
4500	201	Viral Pneumonia	

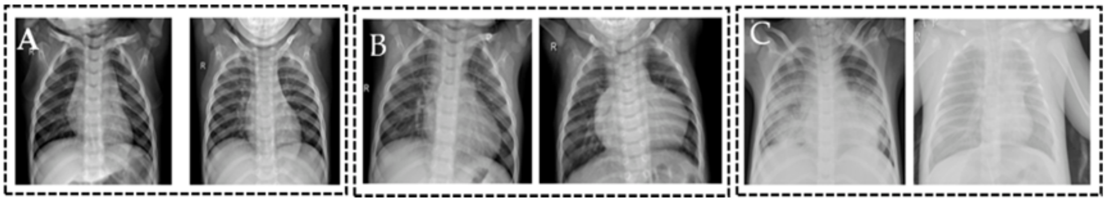


Figure 9. Chest X-rays

Using MATLAB, the researchers in this work taught, assessed, and tested a variety of algorithms (2019a). In Figure 8, we can see the overall methodology of the investigation. After preprocessing the image sets, data is added to them. Then, the training process begins with pre-trained algorithms such as ResNet18, SqueezeNet, DenseNet201, and AlexNet, as well as any methods that the test dataset has been used to evaluate. Every model was trained on a 64-bit version of Windows 10 machine with an Intel© i7-core @3.6GHz CPU, 16GB of RAM, and a 2GB graphics card with a GPU. Table displayed the various parameters utilised to train the CNN models. Although ResNet and DenseNet have numerous variations, this study chose ResNet18 and DenseNet201 since they were easily accessible for Matlab 2019a. Furthermore, in terms of radiographic classification, ResNet18 performs better than previous ResNet models.

Table 3. Various pre-trained CNN models' training settings

Tool	Pre-Trained Models	Dimention	Optimizer	Momentum	Batch size	LR(learning rate)
Matlab (2019a)	AlexNet	227 × 227	Gradient Descent	0.9	16	0.0003
	ResNet18	224 × 224				
	DenseNet201	224 × 224				
	SqueezeNet	227 × 227				

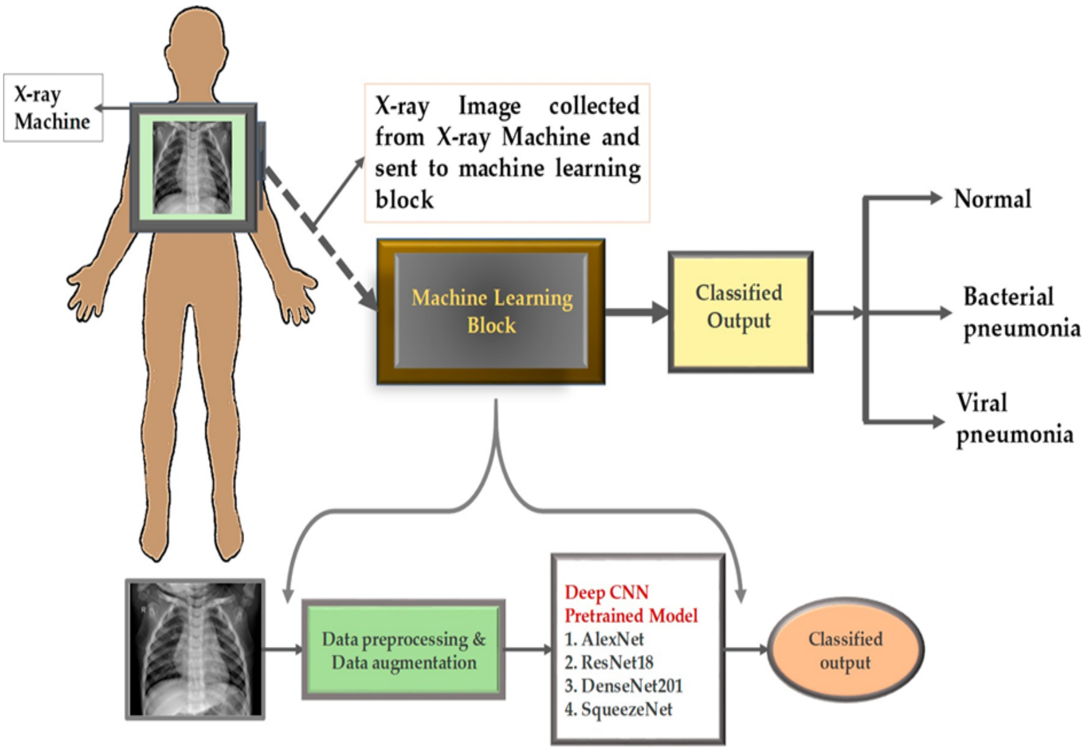


Figure 10. Architecture of SqueezeNe

3.1.1. Preliminary processing

X-ray image resizingAs a crucial step during the preparation of the data procedure, scaling the X-ray pictures was necessary because different algorithms had varying specifications regarding their size. The resized photographs for ResNet18 and DenseNet201 were 224 × 224 pixels, whereas the resized shots for

AlexNet and SqueezeNet were 227 × 227 pixels. Furthermore,Every image was normalised using the pre-trained model's specifications.

3.2. Data Augmentation

CNNs demonstrate their full potential when used to large datasets, as previously said. On the contrary, the functioning database is quite little in size. Utilizing data augmentation techniques to enlarge a restricted dataset is a widely employed tactic during the training of deep learning systems. According to reports, deep learning systems have the ability to improve their classification accuracy by employing data augmentation approaches. Just adding more information to the current dataset will improve the performance of deep learning models without requiring additional data collection. The authors of this work produced additional training sets by developing three innovative augmentation procedures: translation, scaling, and rotation in addition to the existing data augmentation approaches used by numerous deep learning frameworks, as shown in Figure 9.

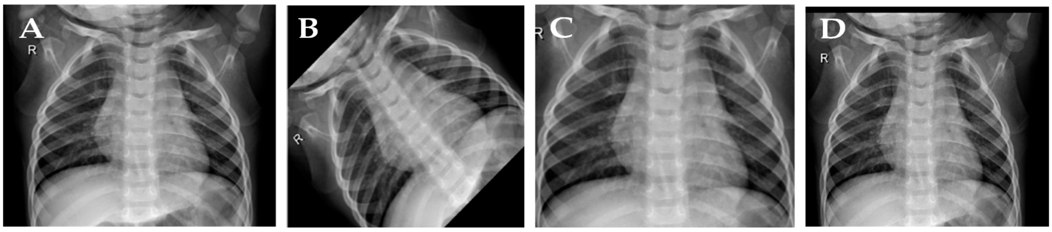


Figure 11. X-ray Chest images: (A) original, (B) rotated, (C) scaled, and (D) translated.

To enhance an image, it is common practice to rotate it clockwise by an angle between zero and three hundred degrees. Each pixel in the image frame is rotated as part of this procedure, which also fills up areas where pixels are absent.. A rotation of 315 degrees (45 degrees anticlockwise) was employed in this work. Another technique for image augmentation is scaling, which is the process of increasing or decreasing the image's frame size. Ten percent of the image magnification was applied overall, as seen in Figure 9C. One can perform image translation by shifting the image in one of two directions: horizontally (width shift), laterally (change in height), or in both directions simultaneously. We translated 10% of the original image horizontally and vertically.

Activation Layer Visualization

By comparing the regions in a picture that exhibited activation in the convolutional layers matching the relevant areas of the original pictures, we were able to examine the image's connecting the convolutional layers to their respective areas in the original images features. Many 2D arrays known as channels make up each layer of a CNN. Several networks were trained on the input image, and the first convolution layer's output activations were analyzed.

Figure 10 displays the activations for several network models. The activation map was normalised to fall between 0 and 1 because it can have a wide range of values. We analysed and contrasted the most significant activation channels with the original image.

Figure displays the activations for several network models. The activation map was normalised to fall between 0 and 1 because it can have a wide range of values. We analysed and contrasted the most significant activation channels with the original image. The study found that these channels respond to edges, with negative effects on dark-left/light-right edges and favourable effects on light-left/dark-right.

Most convolutional neural networks are trained with the primary goal of identifying basic visual components like edges and colours within the initial layer of convolution. Convolutional layers that are deeper are incorporated into the neural network to improve its recognition of increasingly complex objects. Following layers combine characteristics from earlier levels to increase the intricacy of its characteristics. For each model, The activation map in the initial convolutional layers is displayed in Figure 10, the activation channel with the highest power, and the convolutional layer with the greatest depth.

3.3. Different Experiments

In this study, patients with viral pneumonia and those with bacterial pneumonia were divided into two groups. For performance assessments and comparisons, three categories: viral, bacterial, and normal pneumonia were utilised. Through transfer learning, the study also employed four distinct deep learning algorithms.

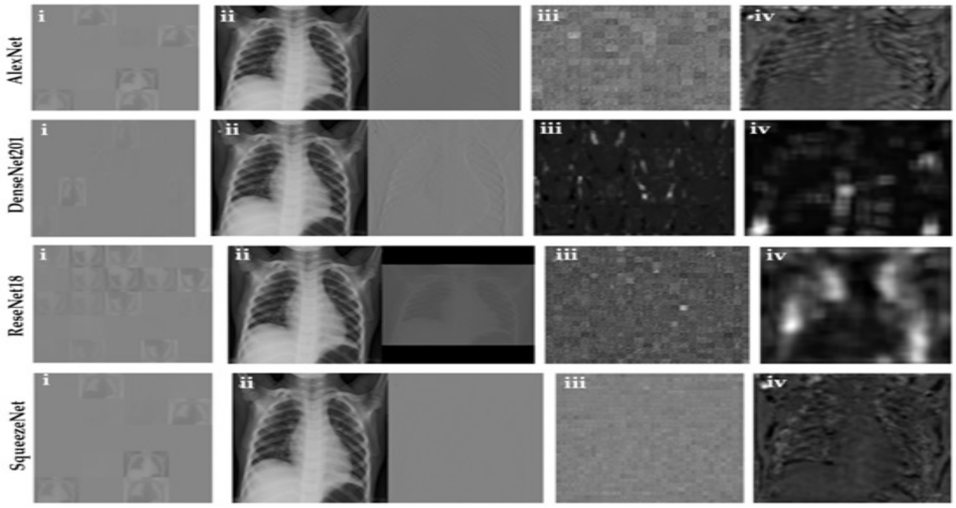


Figure 12. The activation map shows the outcomes of several network models, such as: (i) the 1st-layer of convolution, (ii) the specific channel with the highest activation, (iii) a collection of pictures from the deeper layer, (iv) the consolidated in the deep convolution layer in a unified pictures margins.

Three phases made up the experiment that was conducted for this study. The dataset was first split into two groups: pneumonia and normal. The dataset was split into three categories in the second step: viral, bacterial, and normal pneumonia. The last stage involved dividing the dataset into two groups: pneumonia caused by viruses and bacteria. Images of common, viral, and bacterial pneumonia were categorised using an end-to-end training technique.

3.4. Performance Matrix for Classification

In this investigation, five-fold cross-validation was employed to train and evaluate four CNNs. After training is finished, The six performance measures—accuracy, sensitivity or recall, specificity, precision (PPV), area under the curve (AUC), and F1 score—are employed to assess and contrast the functionality of various networks for the testing dataset. Six performance indicators for several deep CNNs are displayed below:

$$Accuracy = \frac{(TP + TN)}{(TP + FN) + (FP + TN)}$$

$$Sensitivity = \frac{(TP)}{(TP + FN))}$$

$$Specificity = \frac{(TN)}{(FP + TN)}$$

$$Precision = \frac{(TP)}{(TN + FP)}$$

$$F1\ Score = \frac{(2 * TP)}{(2 * TP + FN + FP)}$$

When classifying patients into normal or pneumonia categories to indicate the number of pneumonia images that were correctly identified as such, normal images that were correctly identified as normal, normal images that were mistakenly labelled as pneumonia, and pneumonia images that were mistakenly labelled as normal, the terms True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) were used, in that order.. In contrast, during the classification of viral and bacterial pneumonia. The numbers of pictures of viral pneumonia that were correctly classified as such, pictures of bacterial pneumonia that were identified as such, and viral pneumonia images that were mistakenly identified as such were indicated, respectively, by the metrics TP, TN, FP, and FN.

4. Results and Discussions

Figure 13 compares the training and testing accuracy of various convolutional neural networks (CNNs) used for categorization schemes. Among the three classification methods, DenseNet201 is yielding superior outcomes in terms of quality for both testing and training. The researchers reported classification accuracy of the percentages were 98% for both pneumonia and normal cases, 93.3% for instances of bacterial and viral pneumonia, and 95% for cases of both types of pneumonia.. AUC sis a critical metric for assessing a classification model's efficacy across many categorization schemes, as shown in Figure 14. Figure 14 makes it even clearer that DenseNet201 performs better in comparison to the other algorithms.

.

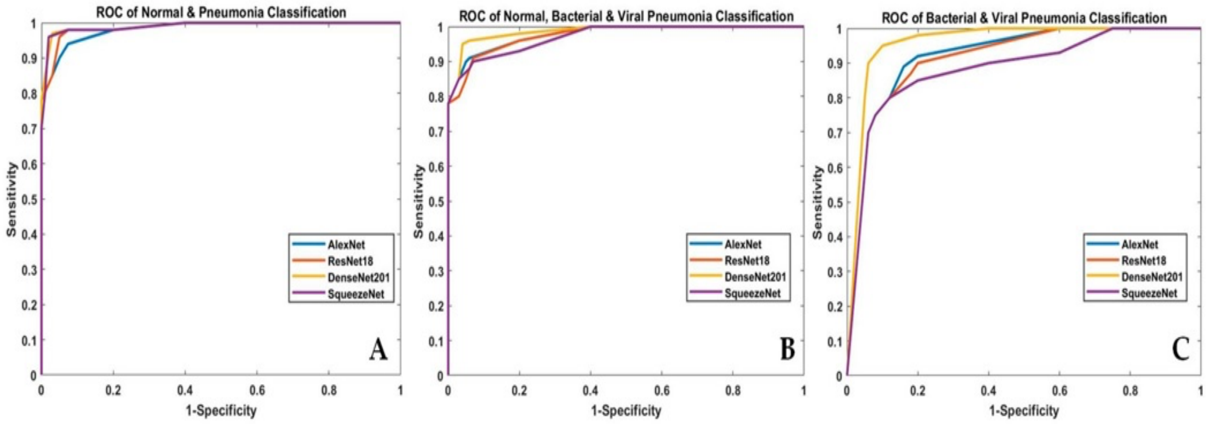


Figure 13. Accuracy in training and testing comparison utilising different models for the classification of CaseA, Case B, and Case C.

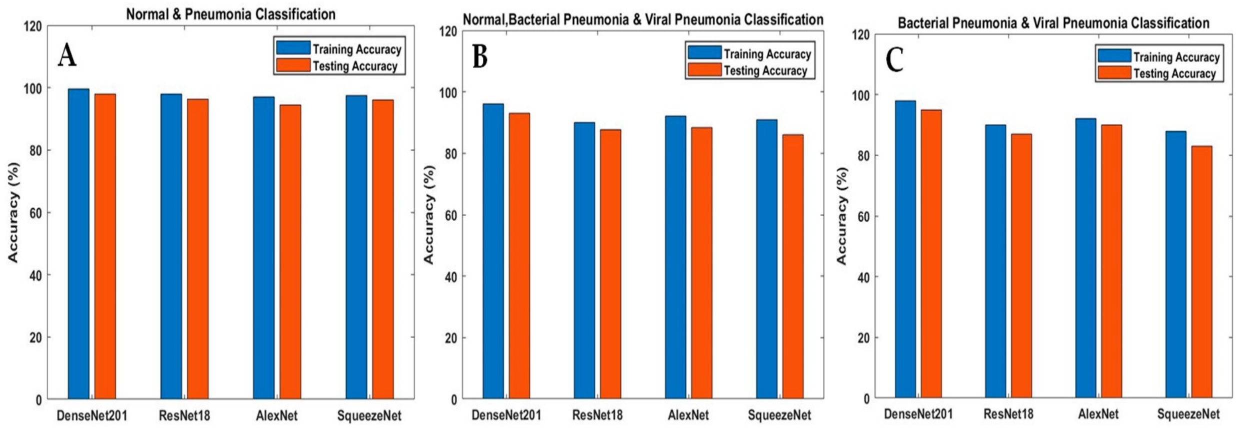
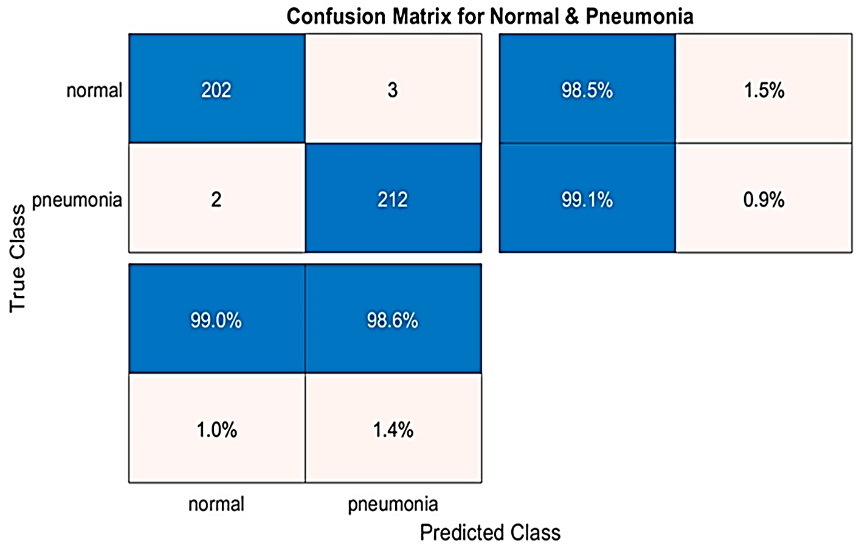


Figure 14. The effectiveness of CNN-based models in differentiating between Case A, normal, Case B , and Case C is assessed through a (ROC) curve comparison.

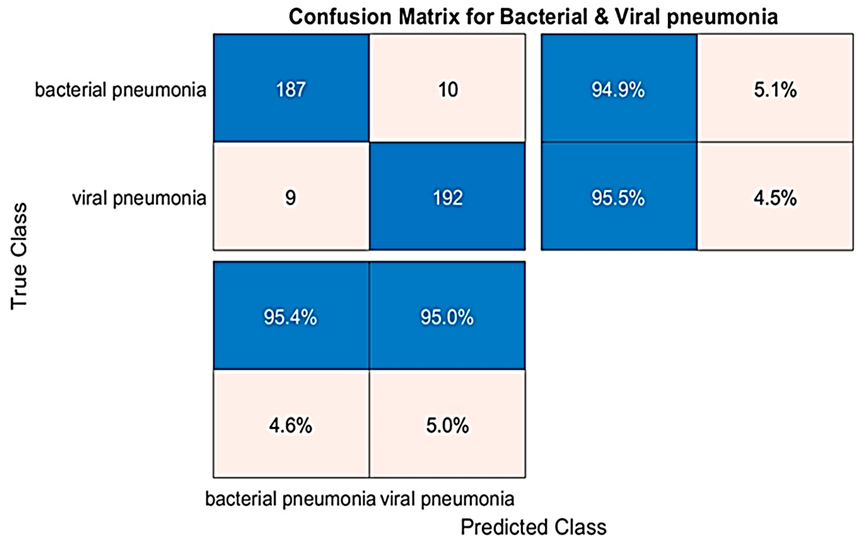
The confusion matrix for the pre-trained Densenet201 model, which performs better, is shown in Figure 13 for three different classes.

Table no 4 presents a concise overview of the performance measures of various convolutional neural network (CNN) algorithms that were assessed for the three distinct classification techniques. DenseNet201 demonstrates superior performance compared to other models across three distinct categorization methods, as seen by several performance indicators.

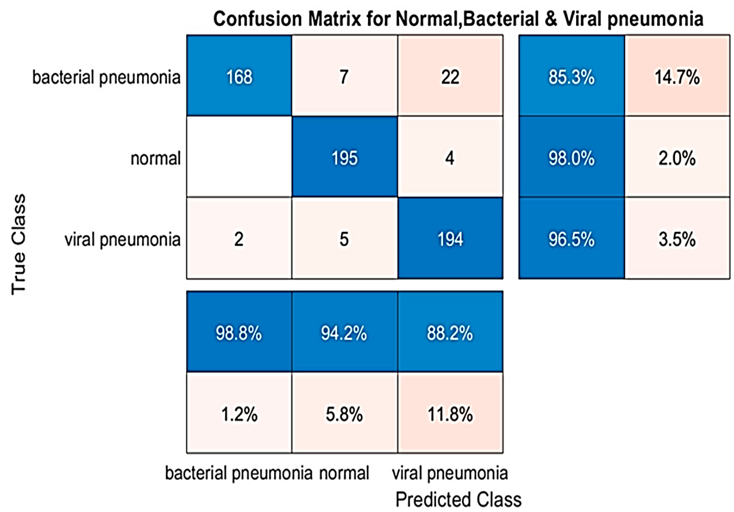
Furthermore, the authors have conducted a comparative analysis of the findings presented in the research with those of other published studies on the identical subject matter. The diagnosis of pneumonia by the use of CXNet-m1, a deep learning system, was documented by Vikash and colleagues[21]. The method demonstrated 99.6% sensitivity, 93.28% precision, and 96.39% accuracy. The efficacy of VGG16 CNNs to differentiate between bacterial, viral, and benign pneumonias was evaluated by Krishnan et al.[34]. Pneumonia diagnosis has a sensitivity of 99.5%, precision of 97.7%, and accuracy of 96.2%. In contrast, the scores for identifying between pneumonia caused by bacteria and viruses were 98.4%, 92%, and 93.6%, respectively. Table 4 presents a comprehensive analysis of research that compares different kinds of pneumonia, as detected using chest X-rays, with previous studies conducted on this topic.



(a)



(b)



(c)

Figure 15. The classification outcomes for using the DenseNet201 model, normal pneumonia, bacterial and viral pneumonia, and normal, bacterial, and viral pneumonia are shown in the confusion matrix.

Table 4. Various deep learning networks use different performance criteria.

Assignment	Pretrained model	Accuracy	Sensitivity (Recall)	Precision	Precision (PPV)	Area under Curve (AUC)	F1 Scores
Normal and Pneumonia	SqueezeNet	0.961	0.94	0.98	0.985	0.96	0.961
	AlexNet	0.945	0.953	0.926	0.931	0.942	0.943
	ResNet18	0.964	0.97	0.95	0.954	0.963	0.965
	DenseNet201	**0.98**	0.99	0.97	0.97	0.98	0.981
Normal, Bacterial, and Viral Pneumonia	AlexNet	0.884	0.883	0.941	0.886	0.911	0.885
	ResNet18	0.877	0.88	0.94	0.875	0.91	0.909
	DenseNet201	**0.933**	0.932	0.967	0.937	0.95	0.935
	Squeeze-Net	0.860	0.858	0.929	0.870	0.893	0.862
Viral and Bacterial Pneumonia	AlexNet	0.90	0.94	0.845	0.86	0.89	0.921
	ResNet18	0.87	0.92	0.82	0.83	0.87	0.873
	DenseNet201	**0.95**	0.96	0.94	0.95	0.952	0.952
	SqueezeNet	0.83	0.905	0.75	0.79	0.83	0.84

Table 5. Contrast with recent comparable studies.

Author	Data	Research Method	Accuracy %
Ozturk et al.[35]	1000 CXR	Yolo - DarkNet	87
Das et al. [36]	1006 CXR	CNN + transfer learning	91.6
Nikolaou et al. [37]	15,153 CXR	EfficientNetB0, TL	95
Alhudjaif et al.[36]	1218 CXR	DenseNet-201	94.9
Srivastav et al.[38]	5856 CXR	DCGAN, CNN, TL	94.5
Singh and Tripathi[39]	5856 CXR	Quaternion CNN	93.7
* Our work	5247 Kaggle Chest X-ray	Pretrained Model CNN	96.9

* The literature's top-performing algorithms are indicated by **, which displays this study's performance..

DesneNet201 is the most accurate prototype currently under development for automatically classifying pneumonia data into three categories: bacterial, viral, and normal. By using an ensemble of pre-trained CNN techniques and a larger dataset for network training, future research may concentrate on improving detection accuracy.esneNet201 exhibits the highest accuracy among current research in creating a model that has automated classification pneumonia data into three categories: viral, bacterial, and normal. Future research could focus on enhancing detection accuracy by using an ensemble of previously learned models and training the network on a bigger dataset on convolutional neural network methods.

5. Conclusions

This paper suggests a transfer learning approach for autonomously detecting pneumonia and its subtypes using deep convolutional neural networks (CNNs).. We conducted training and testing on four widely-used CNN-based deep learning algorithms to differentiate individuals with and without pneumonia based on chest x-ray pictures. Among the four deep CNN networks that were evaluated, DenseNet201 performed to the best possible level. 98% of the photos depicting the right classifications for normal and pneumonia, bacterial and viral pneumonia, and normal, bacterial, and viral pneumonia were made. The precision for the same categories was 97%. Regarding bacterial, viral, and normal pneumonia, as well as bacterial and viral pneumonia, the recall rate was 96%. For both bacterial and viral pneumonia, the specific accuracy was 93.3% and 93.7%, respectively. For viral, bacterial, and normal pneumonia, the accuracy was 93.2%. For viral, bacterial, and typical pneumonia, the recall rate was 99%. Annually, this disease has the potential to cause fatalities in millions of youngsters. An accurate diagnosis of the illness, along with a prompt intervention and appropriate treatment regimen, has the potential to prevent numerous fatalities. During periods of global or developing country medical crises, when healthcare professionals are inundated, Computer-aided diagnosis, or CAD, is a life-saving technique. Furthermore, the input pictures that came from the X-ray equipment exhibit significant inconsistency as a result of the wide range of expertise levels among radiologists. When trained on a small dataset including complex data, like photos, the DenseNet201 model performs remarkably well in diagnosing pneumonia cases, displaying decreased bias and improved generalisation. We are sure that the radiologist will be able to obtain more medically useful images and quickly identify cases of pneumonia with our computer-aided diagnostic tool.

This CAD technique can be applied in various settings, such as screening pneumonia patients at airports, due to its quick classification.

Author Contributions: The conceptual framework presented in this study originated from the collaborative efforts of Memoona Shakeel and Dr Ahmad Naeem, with Memoona Shakeel leading the formulation of the theory and execution of calculations. The validation of analytical procedures fell under the purview of Memoona shakeel and Muhammad Qasim Shafiq. Significantly, Dr Ahmad Naeem played a vital role by encouraging Memoona Shakeel to delve into a specific area, and overseeing the study's outcomes. Each author, including Dr Naeem Aslam, actively contributed to the final manuscript, having extensive conversations on the results during the whole investigation process.
Conflicts of Interest: No conflicts of interest.

References

1. Alishba, I. Training a CNN to detect Pneumonia. 2019 22 december 2023]; Available from: https://medium.datadriveninvestor.com/training-a-cnn-to-detect-pneumonia-c42a44101deb.
2. Aydoğdu, M., et al., Mortality prediction in community-acquired pneumonia requiring mechanical ventilation; values of pneumonia and intensive care unit severity scores. (2980-3187 (Electronic)).
3. WHO, Standardization of Interpretation of Chest Radiographs for the Diagnosis of Pneumonia in Children. World Health Organization: Geneva, Switzerland, 2001.
4. Neuman, M.I., et al., Variability in the interpretation of chest radiographs for the diagnosis of pneumonia in children. (1553-5606 (Electronic)).
5. Davies, H.D., et al., Reliability of the chest radiograph in the diagnosis of lower respiratory infections in young children. (0891-3668 (Print)).
6. Hopstaken, R.M., et al., Inter-observer variation in the interpretation of chest radiographs for pneumonia in community-acquired lower respiratory tract infections. (0009-9260 (Print)).
7. Tahir, A.M., et al., A Systematic Approach to the Design and Characterization of A Smart Insole for Detecting Vertical Ground Reaction Force (vGRF) in Gait Analysis. LID - 10.3390/s20040957 [doi] LID - 957. (1424-8220 (Electronic)).
8. Chowdhury, M.A.-O., et al., Real-Time Smart-Digital Stethoscope System for Heart Diseases Monitoring. LID - 10.3390/s19122781 [doi] LID - 2781. (1424-8220 (Electronic)).
9. Chowdhury, M.A.-O., et al., Wearable Real-Time Heart Attack Detection and Warning System to Reduce Road Accidents. LID - 10.3390/s19122780 [doi] LID - 2780. (1424-8220 (Electronic)).
10. Kallianos, K., et al., How far have we come? Artificial intelligence for chest radiograph interpretation. (1365-229X (Electronic)).
11. Krizhevsky, A., I. Sutskever, and G.E. Hinton, ImageNet classification with deep convolutional neural networks. Commun. ACM, 2017. 60(6): p. 84–90 %U https://doi.org/10.1145/3065386.
12. Ho, K. and J. Gwak, Multiple Feature Integration for Classification of Thoracic Disease in Chest Radiography. Applied Sciences, 2019. 9: p. 4130.
13. Wang, X., et al., ChestX-ray8: Hospital-Scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases, in Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR. 2017: Honolulu, HI, USA. p. 3462–3471.
14. Ronneberger, O., P. Fischer, and T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, in Proceedings of the Lecture Notes in Computer Science. 2015, Springer Science and Business Media LLC: Cham, Switzerland. p. 234–241.
15. Woźniak, M., et al., Small lung nodules detection based on local variance analysis and probabilistic neural network. Computer methods and programs in biomedicine, 2018. 161: p. 173-180 %* Copyright © 2018 Elsevier B.V. All rights reserved.
16. Gu, Y., et al., Automatic lung nodule detection using a 3D deep convolutional neural network combined with a multi-scale prediction strategy in chest CTs. Computers in biology and medicine, 2018. 103: p. 220-231 %* Copyright © 2018. Published by Elsevier Ltd.
17. Jaiswal, A.K., et al., Identifying pneumonia in chest X-rays: a deep learning approach. 2019.
18. Liang, G. and L. Zheng, A transfer learning method with deep residual network for pediatric pneumonia diagnosis. Computer methods and programs in biomedicine, 2020. 187: p. 104964 %* Copyright © 2019 Elsevier B.V. All rights reserved.
19. Souza, J.C., et al., An automatic method for lung segmentation and reconstruction in chest X-ray using deep neural networks. Computer methods and programs in biomedicine, 2019. 177: p. 285-296 %* Copyright © 2019 Elsevier B.V. All rights reserved.
20. Taylor, A.G., C. Mielke, and J. Mongan, Automated detection of moderate and large pneumothorax on frontal chest X-rays using deep convolutional neural networks: A retrospective study. PLoS medicine, 2018. 15(11): p. e1002697.
21. Xu, S., H. Wu, and R. Bie, CXNet-m1: Anomaly Detection on Chest X-Rays with Image-Based Deep Learning. IEEE Access, 2018. PP: p. 1-1.
22. Toğaçar, M., B. Ergen, and Z. Cömert, A Deep Feature Learning Model for Pneumonia Detection Applying a Combination of mRMR Feature Selection and Machine Learning Models. Irbm, 2019.
23. Saraiva, A.A., et al. Models of Learning to Classify X-ray Images for the Detection of Pneumonia using Neural Networks. in Bioimaging (Bristol. Print). 2019.
24. Ayan, E. and H.M. Ünver, Diagnosis of Pneumonia from Chest X-Ray Images Using Deep Learning. 2019. p. 1-5.
25. Khatri, A.J., R. Vashista, H. Mittal, N. Ranjan, P. Janardhanan, R., Pneumonia Identification in Chest X-ray Images Using EMD. In Internet of Things Applications and Future, 2020: p. 87–98.
26. Virkki, R., et al., Differentiation of bacterial and viral pneumonia in children. Thorax, 2002. 57(5): p. 438-441.
27. Gershgorn, D. The Data that Transformed AI Research—and Possibly the World. 2017 [cited 2023 23 december 2023]; Available from: https://qz.com/1034972/the-data-that-changed-the-direction-of-ai-research-and-possibly-the-world/.
28. DenseNet. Better CNN Model than ResNet. [cited 2023 23 december 2023]; Available from: : http://www.programmersought.com/article/7780717554/
29. ResNet, A. and VGGNet, Inception: Understanding various architectures of Convolutional Networks %U https://cv-tricks.com/cnn/understand-resnet-alexnet-vgg-inception.
30. Lecun, Y., K. Kavukcuoglu, and C. Farabet, Convolutional Networks and Applications in Vision. 2010.

31. Huang, G., et al., Densely Connected Convolutional Networks. 2017.
32. Iandola, F., et al., SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and textless1MB model size. 2016.
33. Mooney, P. Chest X-ray Images (Pneumonia). 2018 [cited 2023 24 december 2023]; Available from: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia/.
34. Rajaraman, S., et al., Visualization and Interpretation of Convolutional Neural Network Predictions in Detecting Pneumonia in Pediatric Chest Radiographs. Applied Sciences, 2018. 8: p. 1715.
35. Al Mamlook, R., S. Chen, and H. Bzizi, Investigation of the performance of Machine Learning Classifiers for Pneumonia Detection in Chest X-ray Images. 2020.
36. Das, A., et al., Automatic COVID-19 Detection from X-Ray images using Ensemble Learning with Convolutional Neural Network. 2020.
37. Nikolaou, V., et al., COVID-19 diagnosis from chest x-rays: developing a simple, fast, and accurate neural network. Health Information Science and Systems, 2021. 9.
38. Srivastav, D., A. Bajpai, and P. Srivastava. Improved classification for pneumonia detection using transfer learning with GAN based synthetic image augmentation. 2021. Data Science and Engineering.
39. Singh, S. and B. Tripathi, Pneumonia classification using quaternion deep learning. Multimedia Tools and Applications, 2022. 81: p. 1-22.