IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

An Enhanced X-Ray Image Classification Model Using Deep Convolutional Neural Networks

Suresh Mohan[1], **Subhashini Narayan**[2]

Independent Researcher, Periyar 7th Street, West Gandhi Nagar, Avadi, Chennai, India[1]

Assistant Professor, SITE, VIT University, Vellore, India[2]

Abstract: The purpose of this research is to present a unique artificial intelligence model for image processing by leveraging data cleaning and optimization techniques in a deep convolutional neural network. This novel trifecta of methods has the potential to improve the performance of deep neural networks in image processing. Priority was given to the often-overlooked and inadequate data cleaning and optimization stages required when constructing a deep neural network. We were able to get positive outcomes by employing this strategy. We put this conceptual framework to the test by comparing the chest X-rays of people with and without pneumonia to look for commonalities. Several performance metrics, including precision, recall, and F1-score, were enhanced, and the true positives and true negatives were both improved. The accuracy improved, and we saw a notable enhancement while employing our proposed framework. All things considered, the outcomes imply that our framework provides superior results without compromising any performance indicators.

Keywords: image classification, convolutional neural network, deep learning, VGG16, Inception v3, Xception

I. INTRODUCTION

This century has seen a meteoric rise in the pervasiveness and significance of computers in all facets of human existence. Modern society is more reliant on computers than any other in human history. Computers have advanced to the point that they are now capable of exhibiting intelligence and have successfully penetrated virtually every field. Furthermore, during the last decade, computing power and storage capacity have gotten so inexpensive that we can store terabytes of data and process it in hours, if not minutes. Because of their massive storage and processing capacities, these computers were able to solve challenging scientific problems. This exponential availability of processing power and storage capacity has immensely benefited Artificial Neural Networks. Data processing has improved, but so has image processing, which was once regarded as a very processing-intensive operation but can now be done with ease. We can now run complicated programs on images to gain insight. Neural networks, which are a complicated and resource-intensive method in and of themselves, may now be run at various levels to construct deep neural networks.

Several fields have benefited from technological advancements besides the computing field. The medical industry is a prime example of this type of key field. For a long time, people have been curious about the hidden workings of the human body. In the past, professionals had to rely on many techniques and technologies; but, over the last century, these tools have become so integrated with computers that, nowadays, computers will be able to process those images and provide physicians with detailed views of the body's interior [1]. Thanks to advancements in artificial intelligence, surgeons now have access to robots that can assist them during even the most intricate surgeries. There have even been cases where the robots performed the entire procedure without human intervention.

Even if we have made great technological strides, we still face a constant threat from pathogens in our everyday lives. They are also constantly evolving and generating numerous human health problems. One of these body parts that are often attacked by pathogens is the respiratory tract. Although respiratory infections are quite frequent, they can quickly become fatal. In the past, pneumonia was a devastating illness that primarily affected the lungs. Recently in 2019, Coronavirus disease (COVID), a virus generated by the coronavirus, has spread to humans all over the world [2]. The COVID epidemic killed a shockingly large number of people over just two years. In addition to pneumonia and COVID, there are a variety of other lung infections [3]. The medical practitioners discovered that these disorders impact the lungs, and this could be recognized to a large extent by X-rays during the preliminary health check.

X-rays have been around for quite some time now. Even if the basic procedure of taking X-ray images has remained the same, the post-processing of X-rays has seen significant development. In today's world, it is possible to collect X-rays and then directly feed them into a computer, where they can be used by a radiologist to make a diagnosis [4]. The processing power of computers has improved to the point that they can now handle large image files. Because of this, a

DOI: 10.17148/IJIREEICE.2023.11718

significant amount of research and practice was devoted to the design of systems that could undertake preliminary diagnosis, and a radiologist would only get involved if the system found any abnormalities in X-rays [5]. This not only improved the accuracy of the diagnosis but also helped radiologists and other medical practitioners save a significant amount of time. It is vitally crucial to undertake research studies with the primary goal of discovering ways to enhance the reliability of the diagnosis of abnormalities in X-rays.

An artificial neural network (ANN) is a type of artificial intelligence that is based on the structure of the brain and is inspired by biological systems. These biological neural networks consisting of neurons are what allow the human brain to function. Similar to how neurons in a human brain are linked to one another, artificial neural networks have many interconnected nodes and are made up of numerous layers, each with its own collection of neurons. This enables computers to mimic the network of neurons that constitutes the human brain, allowing them to comprehend information and make decisions in a manner similar to that of humans. The typical structure of an artificial neural network has three levels. The input layer, as its name suggests, receives data from the user in several predefined formats. The inputs are weighted and the bias is computed by the artificial neural network. The hidden layer comes in between the input and output layers where all the arithmetic is done to discover new details and structures. The input goes through several transformations in the hidden layer, and the results of those transformations are transmitted to the user in the output layer. When it came to dealing with difficult situations, images, or videos, the neural network's single hidden processing layer was not particularly effective. Constructing neural networks that include several hidden layers was necessary to find a solution to this challenge [6]. Multi-layer networks like these are sometimes called deep neural networks (DNN). Each layer may consist of multiple neurons, each of which has its own weights and bias. Each layer possesses the capability to carry out complicated computations, and after those computations are completed, the results are passed on to the subsequent layer so that they can be further processed. The weights of a neural network are chosen at random when it is first created, but they are adjusted over time as the model is trained to make more accurate predictions [7].

Finding patterns in images and classifying images are two of the more common applications that could be applied in the modern world. Some of the most important fields that could benefit from the use of image processing are space exploration, medical imaging, face recognition, fingerprint identification, and the diagnosis of plant diseases. Although neural networks are extremely capable of processing complicated data, their application is constrained by the fact that images exist in a two-dimensional space, and by today's standards, even a "typical" image contains a large number of data pixels. Processing images with such a high pixel count requires a significant amount of computational and processing resources. However, the degree of difficulty in terms of time is the most significant consideration.

When it comes to images, it just increases at an exponential rate. Therefore, a method that is capable of processing images quickly and effectively is necessary [8]. This is accomplished by convolution. After being split up into a number of layers, the images are convoluted in order to get a lower overall pixel intensity. This will result in a reduction in the amount of time required to process the image while maintaining the characteristics of the image. This is done at the convolution layer, which is the fundamental structural component of convolutional neural networks (CNN) [9]. There are three primary categories of layers that make up the CNNs. The first layer, known as the convolution layer, is followed by the pooling layer, and then the fully connected layer [10]. There are a lot of different architectures that can be utilized to build CNNs by combining these layers in a variety of different ways. Inception v3, Xception, and VGG16 are a few well-known networks that are utilized for this.

We investigated a number of research papers and web articles on deep neural networks. Convolutional neural networks, which are utilized for image processing, are a well-known and well-documented subject of research. After carefully examining numerous frameworks and implementing those using online tools, we observed that the performance of some of the approaches may be enhanced. To validate our hypothesis, we developed a computer program and tested it against two popular online datasets. After several execution rounds, we discovered that our proposed approach does actually produce better outcomes. This research paper is the product of our investigation, and it describes our proposed framework as well as the outcomes of our experiment.

This research paper is organized into the following sections: We have discussed the detailed report of the literature survey that was conducted in Section 2. Section 3 provides a detailed overview of the proposed framework. A high-level design of our proposal and a detailed level design of our proposal are provided. This section also covers the phases of input data, preprocessing data, and data cleaning. Section 4 describes the performance metrics used. Section 5 describes the datasets that we used for conducting this experiment. Section 6 describes our experiment and its results in detail. Finally, section 7 contains the conclusions.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

II. LITERATURE SURVEY

We examined a variety of studies that focused on developing an effective framework for recognizing patterns in grayscale images, especially those produced with X-ray technology. In this section, we provide a brief summary of the study we conducted and our understanding of its findings.

The paper authored by Sirazitdinov et al. [11] aims to improve automated detection and localization of pneumonia on chest X-ray images. The authors achieve this by proposing an ensemble of RetinaNet and Mask R-CNN methods. They train the model based on the results of using the FPN architecture, which blends low-resolution and high-resolution semantic data. In addition to that, the non-maximum suppression (NMS) algorithm is also utilized. The authors used a dataset obtained from the Kaggle competition that contained a total of 26684 images. The authors had achieved a recall rate of 0.793 points. The authors conclude that their ensemble approach produces a more effective automated detection.
The paper authored by Meng et al. [12] aims to improve the morphological reconstruction capabilities of fluorescence molecular tomography (FMT). They propose a novel locally connected network which is based on K-nearest neighbor. This method builds the inverse process there by finding the mapping relation and distribution. It contains a fully connected network along with a sub-network. They had successfully implemented the big-source simulation as well as the dual-source simulation. They had achieved an average dice index of 0.74 points. They conclude that theirs is the first study that has achieved such an accurate reconstruction of FMT using machine learning.

The purpose of this research paper written by Li et al. [13] is to assist radiologists in the process of detecting lung cancer using chest X-ray images. The authors propose a new method for detecting lung nodules that is constructed using patch-based multi-resolution CNNs and then apply those CNNs to several fusion methods to classify the images. They achieve an accuracy of 99%, while the number of false positives is 0.2 points. The FAUC was at 0.982 points, and the R-CPM score was at 0.987 points. The researchers concluded that this technique may be implemented in clinical settings.

The objective of the research presented in the paper that was produced by Wu et al. [14] is to achieve a higher level of precision in the image detection of pneumonia using chest X-rays. The authors suggest using a system that is a combination of CNN and random forest. To make the processing more effective, they apply an adaptive median filter to get rid of the noise. After that, the dropout approach is utilized to develop the CNN architecture. The CNN deep activation features are employed by the random forest classifier after that, which is used in conjunction with the GridSearchCV class. They succeeded in bringing the overall recognition rate up to 97%, which was above its previous average. The authors conclude that the suggested ACNN-RF system is more effective than the systems that are currently in place.

The research paper written by Chen et al. [15] outlines an innovative solution to the segmentation of CT images. The authors accomplish this by employing a patch mechanism technique and clearing the backdrop with the use of a three-dimensional network. The spatial features are then extracted afterward. The combination loss approach is utilized to perform gradient optimization, which is then followed by data augmentation. They achieved 93.92% in ACC, 78.26 in PRE, 96.39 in REC, and 75.22 in the DSC approach, which they claim is a good improvement over the existing methods. Additionally, they had reached 96.39 in REC, which is an improvement over the existing methods. The authors conclude that the proposed strategy will work better than what is being done right now.

The topic of classification using hyper parameter optimization is an issue that requires extensive computational analysis, which is investigated in the paper that was written by Zhang et al. [16]. They accomplish this by employing a method of search that relies on a surrogate for the best configuration and by constructing a CNN with multiple levels. Through the use of their suggested nonstationary kernel-based Gaussian surrogate model, they have optimized the hyper parameter configuration. The experimental findings shown earlier demonstrate that the nonstationary assumption poses a non-trivial challenge when attempting to optimize hyper parameters in DNN using Bayesian approaches. They conclude that intuition-based priors can be added to the Bayesian framework to improve the effectiveness of optimization.

The purpose of Dixit et al. [17] paper was to provide doctors with a tool for identifying patients with coronavirus. They suggest a three-step process for carrying out detection in real-time. The first stage is the data preprocessing step, which includes the k-means clustering and feature extraction processes. After that comes optimization, followed by classification with the help of an SVM classifier. They are recommending an optimization for the feature that is based on particle swarm optimization. The authors get a higher f-score, as well as correctness, completeness, and accuracy of 93.34%. The authors conclude that they have developed a reliable and enduring paradigm for the diagnosis of COVID.
The purpose of the research study that was authored by Vinod et al. [18] is to assist the pathologist who is utilizing CT to diagnose chest issues. The authors suggest a framework that makes use of the Deep Covix-Net model, which they believe would result in increased productivity.

DOI: 10.17148/IJIREEICE.2023.11718

They implement this model by making use of GitHub and Kaggle, and the findings indicate that there is an improvement. The precision score is evaluated concerning a variety of performance indicators. When they used X-rays, they achieved an accuracy of 96.8%, and when they used CT images, they achieved an accuracy of 97%. The authors conclude that their suggested model is effective for the image segmentation process.

The objective of the research project that was carried out by Sharifrazi et al. [19] was to provide medical professionals with an effective automatic COVID detection system. The authors do this by constructing a framework that combines the CNN, the SVM, and the Sobel filter. After going through the Sobel filter on the dataset, it is then sent to the CNN, and finally, it is given to the SVM classifier to be evaluated using the ten-fold cross-validation method. They were able to achieve an accuracy of 99.02%, a sensitivity of 100%, and a specificity of 95.23% in their experiment. The authors come to the conclusion that their method, which doesn't use a network that has already been trained, works well for identifying COVID images.

The paper authored by Karnati et al. [20] aims to improve the remote diagnosis of patients using X-rays. The authors suggest a deep CNN that is based on multi-scale data and operates using a six-step process. Using various sampling filters, this model will be able to determine the degree of severity of the diseases. Experiments are run using a variety of datasets that are openly accessible to the public. The authors were able to attain accuracy ranging from 96.01% to 100%, with testing times ranging from 0.202 seconds to 1.28 seconds, which is superior to the 14 existing approaches that the authors assessed. The authors believe that their proposed system improves efficacy.

The paper authored by Halder et al. [21] aims at assisting pathologists in classifying lung nodules using CT images. They achieve this through the utilization of adaptive morphology in conjunction with 2-pathway CNNs and the Gabor filter. The framework that has been suggested, called 2PMorphCNN, is made up of two trainable paths that are made up of different classification techniques. Under different datasets, the authors were successful in achieving an accuracy of 96.85, 96.10, and 95.17, respectively. The value of the ROC curve's area under the curve was 0.9936 points. The authors' observations show that the proposed method for slicing high-resolution CT images works much better than other methods. The objective of the research study authored by Mahmood et al. [22] is to enhance the accuracy of the differentiation between cancerous and benign nodules based on CT scans of the lungs. In order to classify the images, the authors suggest an improved framework using a CNN that is based on AlexNet. In addition to this, the authors implement a number of other kinds of layer ordering and hyperparameters. They are able to achieve a level of accuracy that is 98.7%, a sensitivity that is 98.6%, and a specificity that is 98.9%. The authors arrive at the conclusion that the performance of their framework is superior to that of AlexNet.

The purpose of the research paper authored by Maity et al. [23] is to assist medical professionals by enhancing CAD and the identification of diseases utilizing X-rays. The authors propose an automated computer-aided design (CAD) system based on deep neural networks. This system would be able to segment images based on anterior-posterior and posterior-anterior views of the images. The architecture takes advantage of residual connections as well as data preparation techniques including Top-Hat Bottom-Hat Transform and CLA-Histogram approaches. They end up with a DSC score of 0.982 and a JSC score of 0.967 points. The authors conclude that the proposed model is not only lightweight and simple to implement, but that it also outperforms other methods that are already in use.

The paper authored by Aslan et al. [24] aims to classify chest CT images of COVID disease. The authors do this by putting forth a new approach for the classification of the images through the use of segmentation and Bayesian optimization, which determines the hyper parameters. The process begins with the segmentation of the image and then moves on to the augmentation of the data. After that, these images are fed into five distinct CNN models and analyzed. After that, the SVM, k-NN, NB, and DT are provided with this output. The Bayesian Optimization method is used to determine the classification that is the most accurate. Using SVM, the authors were able to attain an accuracy of 96.29%. Sensitivity equals 0.9642, accuracy equals 0.9642, and specificity is 0.9641. The authors conclude that an increase in the number of hyper parameters can produce successful results.

The paper authored by Jalali et al. [25] aims to help diagnose and monitor COVID through the use of image patterns in chest X-rays. The deep learning models are used to do automatic pattern recognition in X-rays, which allows the authors to accomplish this goal. The authors suggest a CNN technique with a KNN classifier implemented in place of the softmax layer as the very final layer. In addition to this, they suggest an enhancement for the swarm optimizer. They employ the Cauchy mutation, EBCH, and tent chaotic map methods during the search. The authors conclude that they achieved improved performance.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

III. PROPOSED METHODOLOGY

According to the findings from existing studies, an ensemble of deep learning models, including VGG16 [26], Inception v3 [27], & Xception [28] was used to detect variations in the input images. The collective accuracy obtained was around 78 percentage. This value is the average of the accuracy values that was obtained from the input dataset of lung X-ray images to detect pneumonia. This accuracy level is good, yet we tried to improve the accuracy levels by proposing an improved model for detecting even the smallest deviation from the ideal requirement.

The proposed model uses the deep neural network to optimize the performance of the network [29]. Preprocessing is done on the input dataset to remove incorrect and irrelevant data, duplicates, outliers, etc. [30]. Numerous studies suggest that thoroughly cleaning the input data would result in significant performance improvement [31]. Subsequently, we cleaned the data [32]. It would ensure that only standardized, high-quality data would enter our system.

Then we do the feature selection and data validation. After validation, the dataset is split into training and testing models. The training model is then fed into the deep neural network system. Optimization is used to improve the performance of the deep neural network [33]. It enables us to improve the accuracy [34]. Then the testing model is applied to the deep neural network system. The system classifies the testing model based on its learning from the training model [35]. The output is then sent for performance evaluation to see if this model provides any improvements over the existing methods. The Fig. 1 High level design of the proposed model provides a high-level design of the proposed framework.

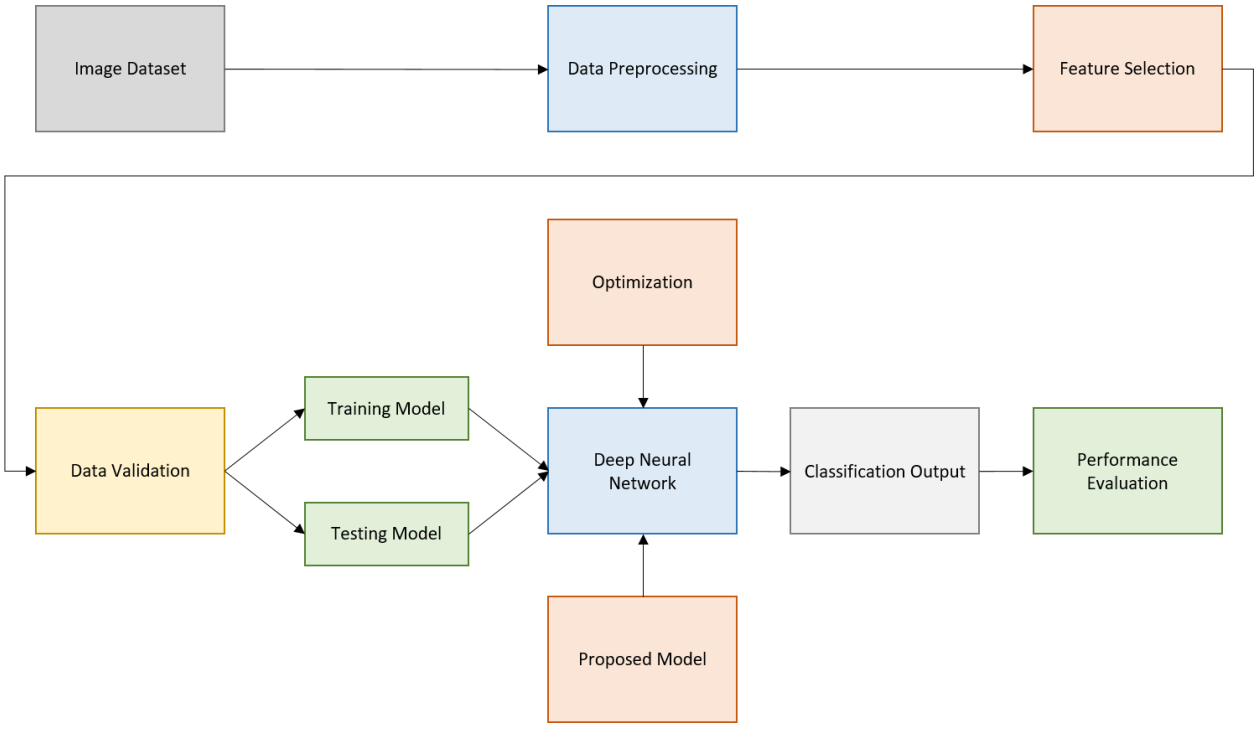

Fig. 1 High level design of the proposed model

The primary intention of this proposed framework is to improve the performance measures of the existing methods. We focus on two areas that significantly improve the performance measures. The data cleaning layer. Even though this seems trivial and could be done with any method, carefully choosing an appropriate method for particular use cases would provide cleaner input training datasets, thereby increasing the efficiency of the classification. The other focus is on the optimization technique. We optimize the existing methods and the deep neural network using the optimization method. The Fig. 2 Detailed level design of the proposed model provides a detailed design of the proposed framework.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

Data is the most essential component of this method. We need a significant amount of data in order for the model to be able to train itself and improve its accuracy, but before it can do either of those things, it can't make any predictions. We should start by providing the model with training data when we initially expose it to the data. After the model has been trained, we should continue to provide it with test data so that we can evaluate how well the model is performing its tasks. After we have determined how well the model is functioning, we will be in a position to provide it with the test data.

It is of the utmost importance to measure and analyze the performance of the model because doing so will tell us whether or not the model accurately predicted the results and whether or not we can rely on these results. We will initially provide the test data and then provide the test data after the model has learned.

We prepare the data for modeling by lowering the image dimensions as part of data cleansing. The size of the image input will differ considerably. It introduces inconsistency into the model, making the identification of patterns more challenging. Hence, dimensionality reduction is performed to normalize the image's size and dimensions. We've chosen 150 by 150 pixels. Therefore, we apply methods to reduce the image to 150 by 150 pixels without significantly affecting the quality of the dataset. It would help our dataset be more consistent. Reducing the dimensions allows for fewer processing parameters and a simpler structure. This operation needs to be done on the training dataset, the validation dataset, and the test dataset.

The primary focus of this study is on grayscale images. The image's intensity, which runs from deep black to brilliant white and all shades of gray in between, is encoded in each pixel. The minimum and maximum values are scaled between 0 and 1, respectively, and the image's intensity can be expressed as a range from deep black to brilliant white. As a result, we will be able to process the images with the intensity of the regions where the actual deviations should be identified more effectively.

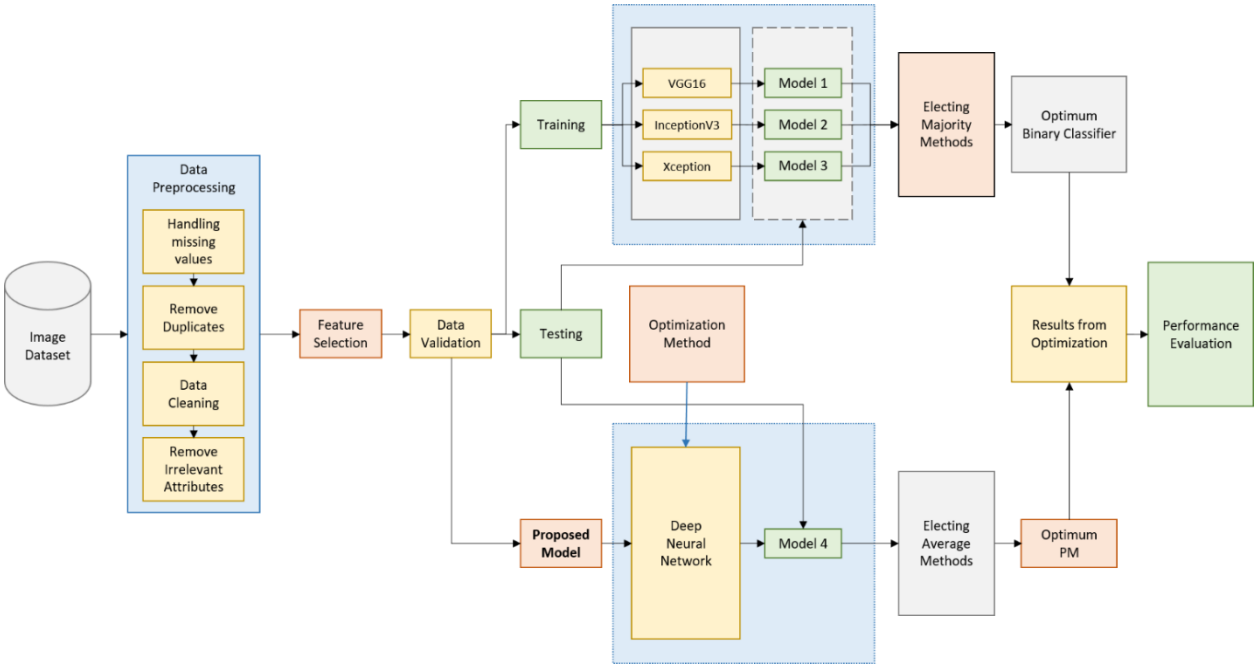

Fig. 2 Detailed level design of the proposed model

We'll need images with and without the target regions to train the model. The model will distinguish between the target regions in these images using training data. The model can then classify images based on the label and the intensity value of each pixel, which will range between 0 and 1 depending on the learning environment. Using this strategy, the model will be able to label images based on the region of interest and then classify images using the self-learned label.

As we are combining multiple data sources, it is highly possible that the data contains duplicate and/or mislabeled values. In fact within the same dataset also there are high chances that there are duplicate and mislabeled values. So, the collected data should be now cleaned of any duplicates, incorrect data, incomplete data, or corrupted data. To identify duplicates, we could use the Levenshtein distance method.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

The difference between the two sequences could be identified using this method using which we could find out if there are any duplicates within our data and can remove them. Also, this method can identify the steps required to change one value into another value. We could employ this method to transform data from the outliers into valuable data so that we get a more sophisticated data set. The steps in data cleaning are depicted in Fig. 3 Data cleaning.

This method computes the minimum number of edits needed to transform strings. We could do inserts, deletes, and substitutes using this method. Here every operation has a unit cost.

$$lev_{x,y}(|x|, |y|) \tag{1}$$

where,

$$if\ \min(a, b) = 0,$$

Otherwise,

$$lev_{x,y}(a, b) = \begin{cases} \max(a, b) \\ \min \begin{cases} lev_{x,y}(a-1, b) + 1 \\ lev_{x,y}(a, b-1) + 1 \\ lev_{x,y}(a-1, b-1) + 1_{(x_a \neq y_b)} \end{cases} \end{cases} \tag{2}$$

The Indicator function is equal to 1 in the above equations, except when $x_a == y_b$. In such cases, it will 0. The length of the string x is denoted by $|x|$. The value between the first a characters of the string x and the first b characters of the string y is denoted by $lev_{x,y}(a, b)$.

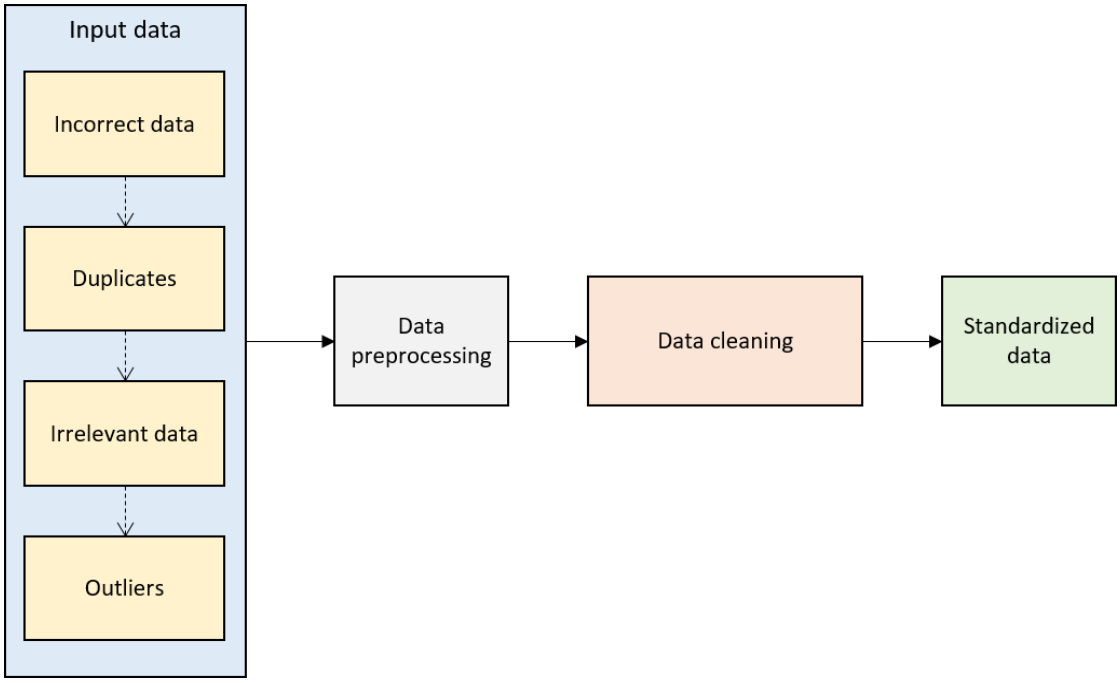

Fig. 3 Data cleaning

We will use the above method to find the value between the strings thereby finding the data that are related to each other and might be direct duplicate. To identify this, we can use the following algorithm and implement it in the computer program. The following algorithm is used for data cleaning.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

Algorithm 1: Data cleaning

```
Method COMPUTE (x,y)
If the length of x is equal to 0 then
        Return the length of y.
If the length of y is equal to 0 then
        Return the length of x.
If length of x and y is not equal.
        INDICATOR = 0
Else
        INDICATOR = 1
INDICATOR is equal to 0 if l else,
Calculate
        substring of x (INDICATOR), y
        substring of y (INDICATOR), x
        substring of x (INDICATOR), y (INDICATOR)
Return the minimum of the above three values which is the COMPUTE.
End.
```

An optimization algorithm finds an optimum or satisfactory solution for a problem by executing iteratively. This is used to find a desirable result for the problems that we are addressing and trying to bring an improved result. Ant colony optimization is a type of optimization algorithm using probabilistic technique. This method is highly based on the working principle of communication in nature. For A to communicate with B, it has to move closer to each other. It will enable the best local communication and facilitate group communication by transiting information. A vector generally move from node x to node y with the following probability

$$P_{x,y} = \frac{(\tau_{x,y}^{\alpha})(\eta_{x,y}^{\beta})}{\sum(\tau_{x,y}^{\alpha})(\eta_{x,y}^{\beta})} \tag{3}$$

$$\tau_{x,y} = (1-\rho)\tau_{x,y} + \Delta\tau_{x,y} \tag{4}$$

$$\Delta\tau_{x,y}^{n} = \begin{cases} \frac{1}{L_n} \\ 0 \end{cases} \tag{5}$$

$$\tau_{xy} \leftarrow (1-\rho).\Delta\tau_{xy} + \sum_{n}^{m} \Delta\tau_{xy}^{n} \tag{6}$$

$$\tau_{xy} = (1-\varphi).\tau_{xy} + \varphi.\tau_0 \tag{7}$$

$$\tau_{xy} \leftarrow (1-\rho).\tau_{xy} + \rho.\Delta\tau_{xy} \tag{8}$$

Where, x, y is the edge, $\tau_{x,y}$is the path, α, β are the parameters to control the path and P is the probability. The amount of path will vary and is computed as given above. As per this model, the path on a specific node is directly proportional to the number of traversals through this path in the past. In the following algorithm for optimization, we find the best path between x and y.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

Algorithm 2: Optimization algorithm

```
Method Optimization (x,y)
For every element in x and y iterate the following
        Set the values for x and y as 0
        Iterate for every element in y from 1 to n
        Find probability to visit each node.
        Select a node as per probability value.
        If the element could traverse the path
                Return.
        Else go into the selected node and
                begin iteration again.
Save and rerun the loop for n times.
Return the best value.
End.
```

IV. PERFORMANCE METRICS

The learning curve is a graphical representation of the relationship between two factors that are related to each other. In this study, we have used the learning curve to represent the accuracy of the learning method and the number of epochs that have been carried out for the purpose of achieving the goal. The accuracy is measured on the vertical axis, which usually increases with the increase in the epoch, which is measured on the horizontal axis. It is usually interpreted as the greater the number of epochs that are applied to the learning method, the greater the level of accuracy that is achieved. Nevertheless, we were able to determine that this particular curve does not exhibit constant growth. Rather, after a certain point, there was little impact on the accuracy brought about by increasing the number of epochs. In spite of the fact that our first assumption was that we would have a steep learning curve, we discovered through actual execution that our learning curve would be gentler. Based on our findings, we came to the conclusion that increasing the epochs would not result in a significant improvement in the accuracy. Furthermore, after a certain point, it becomes less efficient to run a higher number of epochs. The second observation concerns the angle of the slope (gradient) of the curve. It showed no dependence on the overall performance of the learning method. It expresses the expected rate of increase in performance as the number of epoch increases. We have provided a more in-depth description in the experiments section.

The effectiveness of a learning method can be analyzed through the use of a table called a confusion matrix. A confusion matrix is a useful tool for visualizing and summarizing performance data. It is a simple tabular representation that is capable of indicating a diverse assortment of characteristics. In this study, we made use of the confusion matrix to illustrate the relationship between the actual true and false outcome values and the predicted true and false outcome values.

A true-positive (TP) is when the actual and predicted outcomes are both true. A false-positive (FP) is when the actual is false and the predicted outcome is true. A false-negative (FN) is when the actual is true and the predicted outcome is false. A true-negative (TN) is when the actual and predicted outcome are both false.

Accuracy is the measure of how many correct predictions were made overall. This is calculated by finding the ratio of total true-positives and true-negatives identified to the sum of observations. Precision is the measure of how many correct positive predictions were made overall. It is calculated by finding the ration of the sum of true positives to the sum of observations that are positive. Sensitivity (Recall) is the measure of how many correct positive predictions were made out of all the positive predictions. It is calculated by finding the ratio of the total true positives to the sum of true positives and false negatives. Specificity is the measure to determine how many correct negatives were made out of all the negatively predicted values. It is calculated by finding the ratio of the total true negatives to the sum of true negatives and false positives. The number of times a specified class actually appears in the dataset is referred to as its support. To determine it, we add up all of the values in the table. The F-measure is the measure to determine the high precision and low recall and vice versa between two data sets by finding the harmonic mean of the combination of the recall and precision. It is calculated by finding the ratio of two times the product of recall and precision to the sum of recall and precision.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

V. DATASET

For our experiment, we used two datasets. The first dataset is the Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification from the Mendeley Data [36]. The Mendeley's pneumonia dataset is made up of 6,000 different chest X-ray images. Of these, 4,500 were taken from people who had pneumonia and another 1,500 were taken from people who were healthy. The dataset was divided up into three parts: the training data, the validation data, and the test data. In each of these three parts, the images consist of chest X-ray images taken from people who were infected with pneumonia as well as people who were healthy.

The CoronaHack chest X-ray images from Kaggle make up the second dataset [37]. There are a total of 5910 chest X-ray images in the CoronaHack dataset. Seventy-three percent came from those with COVID, while the remaining twenty-seven percent came from healthy individuals. The dataset was divided into two parts: training data and test data. Each of these two parts includes chest X-ray images from both healthy people and people infected with COVID.

VI. EXPERIMENTS AND RESULTS

As was just discussed, the accuracy of a deep neural network is highly reliant on a wide variety of different parameters. We propose that the optimization approach be incorporated into the model, and then we will monitor how it affects the overall performance of the model. To verify the validity of our proposition, we implemented this framework. We leveraged the images that were included in the publicly accessible Mendeley's pneumonia dataset and Kaggle CoronaHack dataset available online as described in the earlier section on dataset.

When we put this dataset through the data cleaning process, we discovered that many of the images did not have the correct dimensions; consequently, we had to get rid of the images that could not be recognized. When we were using the Mendeley's pneumonia dataset, the cleaned-up dataset had 3,250 images in total once it was completed. Out of that total, we selected 100 images at random to use as test data for testing this model. The rest of the data, about 85% of it, was used as training data. It included about 2650 images. The remaining 450 images were set aside for use as validation data. The 2650 images that make up the training data set have not yet been assigned any labels. It is expected that the model will be able to label these images as well as identify those images that deviate from the typical image pattern. A similar process was done for the CoronaHack dataset as well.

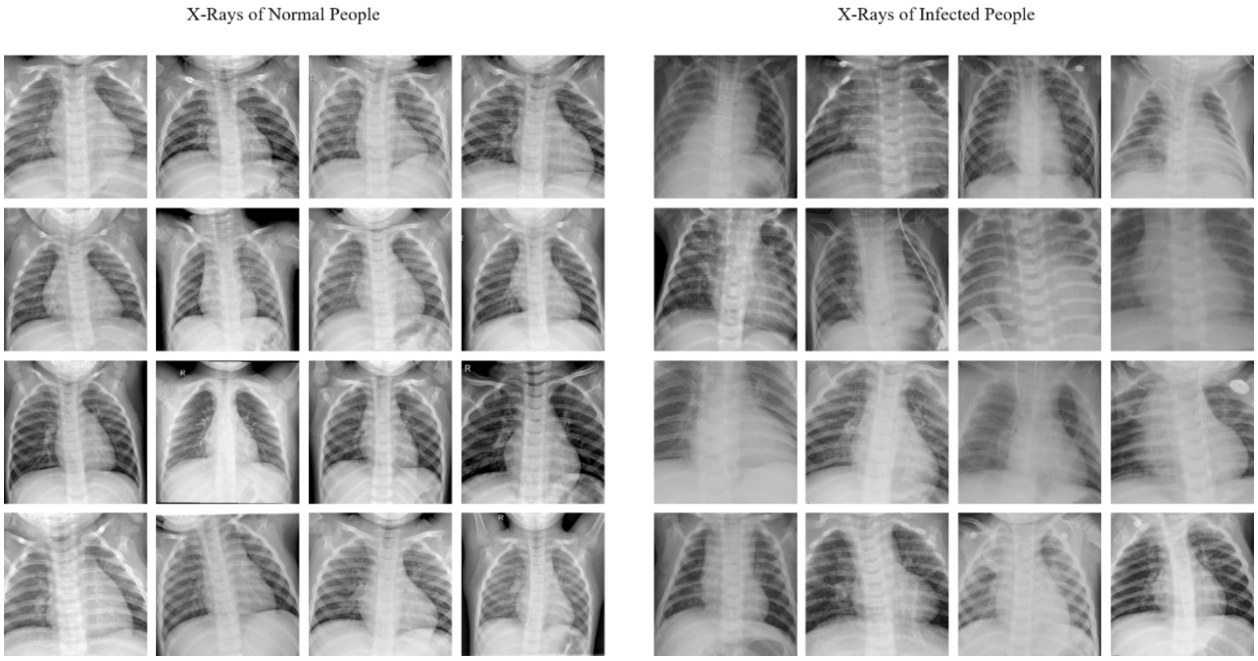

Fig. 4 Chest X-ray images of normal people vs infected patients taken from Mendeley's pneumonia dataset

The dataset consisted of images that are of varying sizes. We normalized the images and scaled them to a standard size of 150 by 150. The images that are used for training and validating would then be ensured to have the same dimensions.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

The Fig. 4 Chest X-ray images of normal people vs infected patients taken from Mendeley's pneumonia dataset contains samples of scaled images. Additionally, providing images of reduced size will have a load-reducing effect on the model while still preserving the features, which will also ensure that the results are unaffected. This is very important for the image classifiers since it would prevent them from having large images in the dataset, which is something that might significantly impact the performance of this evaluation.

There are two labels identified, with 0 being normal and 1 being infected with pneumonia. Based on this information, the weight of the class was also determined. It is important to avoid having any kind of bias in the class weight to avoid any erroneous classification output. Based on the input dataset, we assigned the class weight to both the minor and major classes. This will affect the execution of the model when it is being trained, which in turn will affect the classification. The value we assigned to the minor class was much greater than the value we assigned to the main class. This will normalize during the training phase towards a median value, leading to more appropriate results. A similar process was done for the CoronaHack dataset as well.

To evaluate this strategy, we built a convolutional network using the VGG16 architecture [38]. This state-of-the-art method employs a 16-layer deep neural network and serves as a standard for evaluating datasets with an image focus. Each layer of this method has weights associated with it [39]. We provided a distinct weight for each of these layers. However, we kept these weights standard when we implemented our proposed model, thereby making sure the difference in weight is not affecting the results and, thus, the performance of the proposed method. In all, there are 16 layers in the VGG16. The VGG16 runs through 16 layers. First is the input layer. Then two convolutional layers Then a pooling layer. Then two convolutional layers and one polling layer. Then three convolutional layers and one pooling layer are run three times. Then the results are flattened and a dense output is produced. A similar convolutional network was built using the Inception v3 [40] and Xception [41] architecture as well.

Epochs are the number of times the training dataset is processed by the algorithm throughout its execution. The outcomes of each epoch are different, and to get the best possible outcomes, we often take the average performance of all of these epochs into account. This topic is covered in further depth in subsequent sections. Each time one of these epochs occurs, the model will conduct an update on its internal parameters, potentially leading to an improvement in performance. During the execution of the method, both the benchmarked technique and our suggested approach provided us with a total of 15 epochs. This will ensure consistency and possibly not affect the outcomes of the executions.

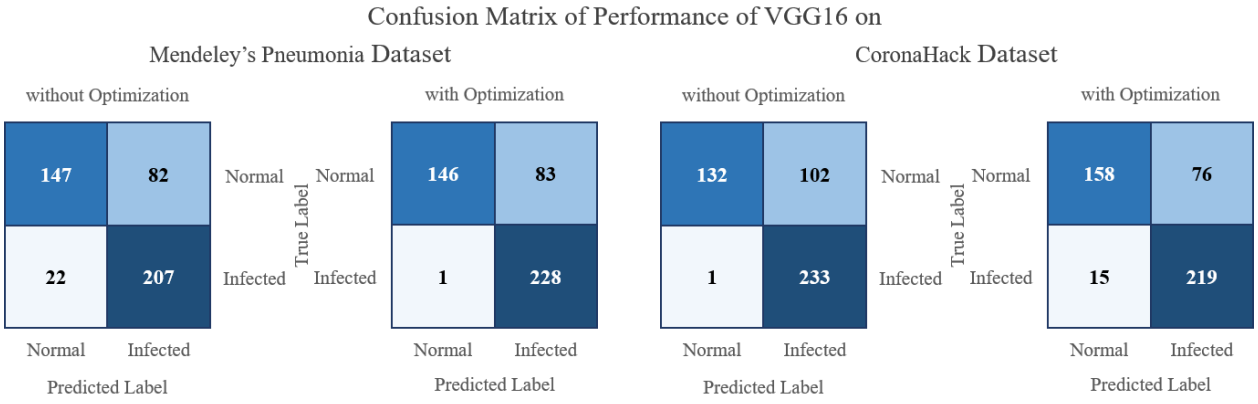

Fig. 5 Confusion matrix of the performance of VGG16 on Mendeley's pneumonia dataset and CoronaHack dataset

The confusion matrix is an illustration of the output of the suggested approach in comparison to the method that is currently being used. When experimented on the Mendeley's pneumonia dataset, the current approach had a success rate of 64.2% when it came to correctly identifying normal instances, whereas it had a success rate of 90.4% when it came to correctly identifying infected cases. Having said that, the model showed signs of improvement once the optimization had been carried out. It was successful in identifying normal cases 63.8% of the time, whereas it was successful in identifying infected cases 99.6% of the time. This is a performance increase of 9.2% for infected cases. A similar result was obtained using the CoronaHack dataset as well as depicted in Fig. 5 Confusion matrix of the performance of VGG16 on Mendeley's pneumonia dataset and CoronaHack dataset.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

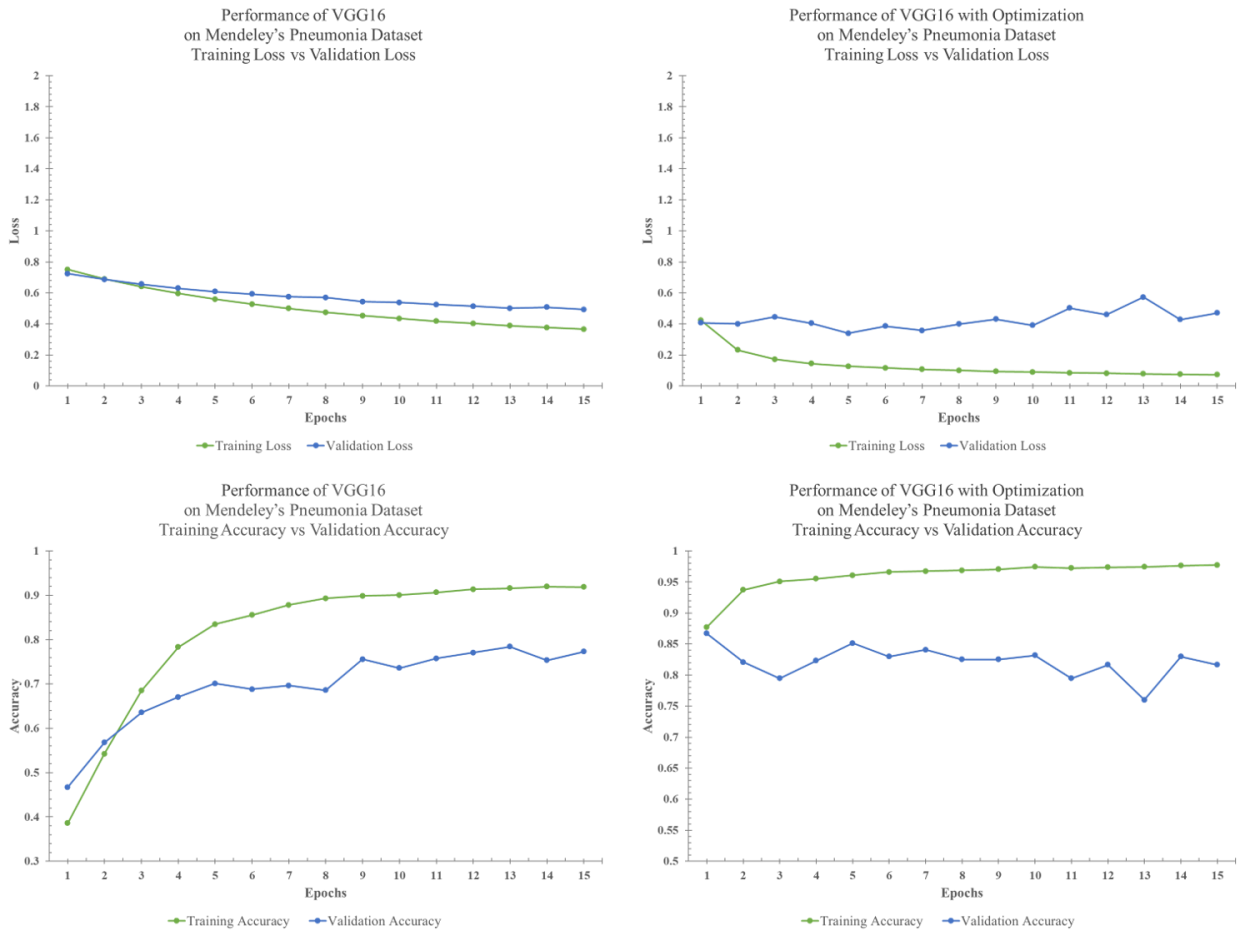

Fig. 6 Accuracy and loss performance of VGG16 on Mendeley's pneumonia dataset

For the benchmark methodologies we have available currently, we can see how the training loss compares to the validation loss in the Fig. 6 Accuracy and loss performance of VGG16 on Mendeley's pneumonia dataset.

We can observe that there is a trend toward improvement in the training loss as the number of epochs increases from one to fifteen, but there is a trend toward degradation in the validation loss. During the first epoch, the training loss was 0.7507, but after that, it rapidly dropped till it reached 0.3657 points. It is a significant increase in the amount of training gained during a span of fifteen epochs. We can observe that the validation loss began at 0.7232, and up to epoch 10, the validation loss was within a limit that was considered acceptable. After that, however, there was a discernible decline in its condition, and it reached 0.4925 points. In general, we can observe that the loss is now around 0.2307 points higher. When we utilized our proposed optimization method and ran the code, we saw that the validation loss did not improve considerably over epochs. The proposed method began with 0.4227 points as the starting value for the training loss, which is a weaker starting point than the benchmark methods. Despite this, the training loss showed steady progress over the iterations, and by the 15th epoch, it had decreased to 0.0724 points. It is a very substantial improvement in comparison to the current approach, which had a value of 0.3657 at the end of the 15th epoch. However, the situation was somewhat different with regard to the validation loss. The validation loss for our proposed method was 0.4068 at the beginning of the first epoch, and it subsequently got lower and lower as each epoch progressed. The validation loss ended up being 0.4701 after the 15th epoch, which is a slightly above the validation loss that we obtained when we did not use the proposed optimizer. While the validation loss was 0.4925 without the optimizer, it was just 0.4701 with the optimizer. It is a difference of 0.0224 points in comparison to the existing methods. The Fig. 6 Accuracy and loss performance of VGG16 on Mendeley's pneumonia dataset also illustrates a comparison between the training accuracy and validation accuracy. The accuracy of the training started off at 0.3858 during the first epoch, and it steadily grew to 0.8783 at the end of the seventh epoch. After that, the development was mostly linear, and by the time the 15th epoch was completed, the training accuracy had reached 0.9185. During the first epoch, the validation accuracy was found to be 0.4672. The validation accuracy was quite inconsistent during the progression of the epochs. It appeared like it was moving back and

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

forth between 0.65 and 0.80. It fluctuated throughout each and every cycle of the epoch. However, during the 13th epoch, the validation accuracy began to show signs of a gradual drop, eventually reaching a value of 0.7729. The findings were different after the optimization method that had been presented had been implemented. The accuracy of the training was measured at 0.8772 during the first epoch, and it continued to rise until the fourth epoch, when it was measured at 0.9550. After that, the progress was not considerable during the subsequent eleven epochs; it had improved by a meager 0.0223 points, ultimately culminating in a score of 0.9773. The validation accuracy began at 0.8668 and continued to progressively fall until the 3rd epoch, when it reached a value of 0.7948. In the fourteenth epoch, it had a significant improvement of 0.0699 points. However, during the subsequent epoch, this decreased to the typical level, and the development pattern remained linear. There was a brief decrease in the validation accuracy during the 13th cycle, but it immediately returned to its previous level during the subsequent cycle, and by the conclusion of the 15th epoch, it had reached 0.8166. Compared to the previous method, which had a value of 0.7729 at the end of the 15th epoch, we now have a value of 0.8166, which is a good improvement.

The training loss before and after using the proposed method is shown in Fig. 7 Performance of VGG16 with and without optimization on Mendeley's pneumonia dataset. We see that the training loss began at 0.7507, progressed smoothly rapidly till the end of 15 epochs, during which the score was 0.3657 points. However, this performance was superior after the optimization was applied. The training loss began at 0.4227 and, after being measured over the course of 15 epochs, it was found to have reached a final value of 0.0724. We see that even after 15 epochs, the score hovered around 0.0724, which is way better than the training loss of 0.4227 at the beginning of the first epoch in earlier methods. The training loss was significantly higher compared to the earlier methods. In contrast, the validation loss, on the other hand, showed signs of improvement as the epochs progressed. The validation loss for the existing methods began at 0.7232 and fluctuated between 0.45 and 0.70 for each epoch. Nevertheless, after all the 15th epoch, the score had gradually dropped, and it finally settled at 0.4925. Following the implementation of the optimization, the findings showed no better results. The starting loss was 0.4068, which is lesser than the greatest score achieved in any of the epochs preceding the epoch of the optimization. On the other hand, the score continued to drop with the passing of each epoch, and by the time the 15th epoch was over, it had reached a value of 0.4701 points.

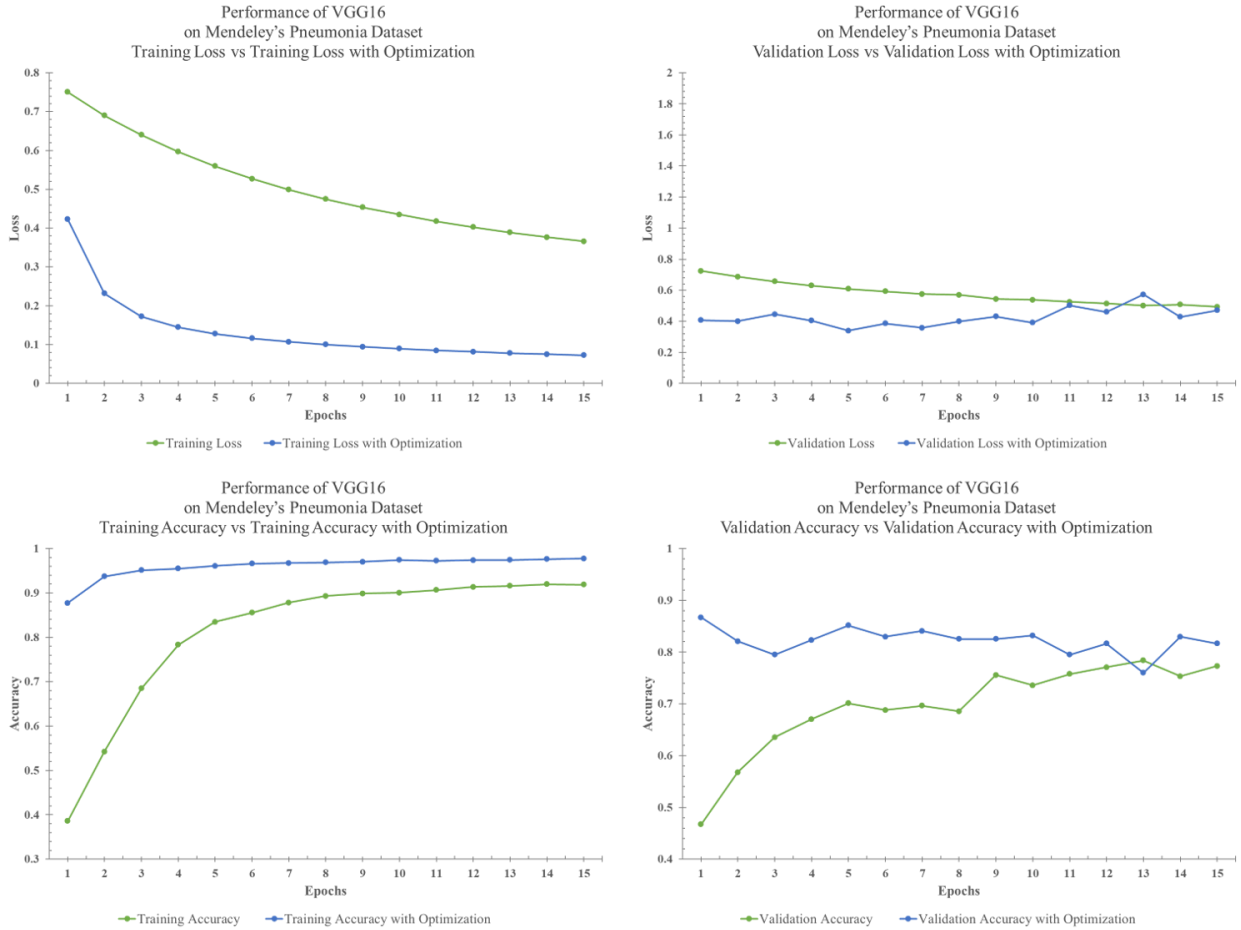

Fig. 7 Performance of VGG16 with and without optimization on Mendeley's pneumonia dataset

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

The trajectory that the training accuracy took both before and after the optimization was implemented was considerably different from the trajectory that the training loss took. In the beginning, we saw that it began at 0.3858 points, and then after that, it grew steadily for the entirety of the cycle of epoch. The score was 0.9185 by the time the 15th epoch was over and done with. This demonstrated that the procedures that were already in place were more effective during the training phase; nevertheless, the findings from the validation phase showed that we require further improvements in that particular domain. Following the implementation of the optimization approach, the training accuracy began at 0.8772 at the beginning of the epoch and continued to rise at a linear rate up to the eighth epoch, when it reached 0.9688. After that, however, the pace of growth slowed down, and after the subsequent seven epochs, we witnessed an improvement of just 0.0085 points.

When compared to the validation accuracy, which produced more positive outcomes when utilizing the optimization methods, this finding is a stark difference. For the existing methods, the validation accuracy started at 0.4672 and then oscillated between 0.65 and 0.75 for the rest of the validation phase. The score ended up being 0.7729 at the end of the 15th epoch, which is only 0.0720 points away from when it was at fifth epoch. Following the implementation of the optimization method, the results indicated a meager amount of progress. At the very beginning, the score was at 0.8668, which is a significant drop from the score that it had been before the optimization procedures were used. On the other hand, the rise was not so consistent throughout each epoch, and we saw a decrease up to the third epoch, when the score reached 0.7948. After that point, however, the growth rate slowed down, and it became weak over time. There was a drop in performance during the consecutive epochs. The final score was 0.8166 after 15 epochs elapsed. We were able to achieve a similar kind of results when we used the CoronaHack dataset. This is depicted in Fig. 8 Performance of VGG16 with and without optimization on CoronaHack dataset.

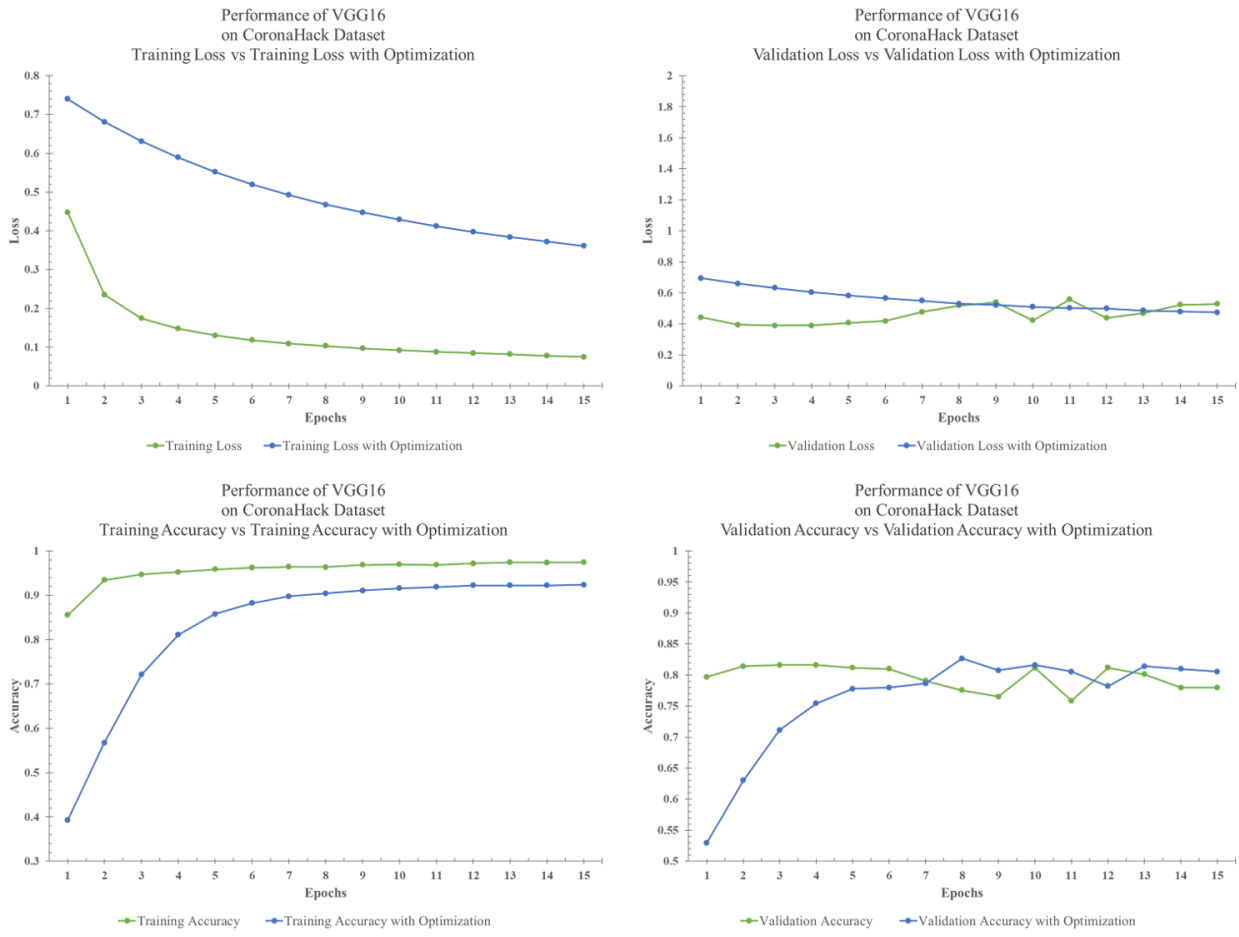

Fig. 8 Performance of VGG16 with and without optimization on CoronaHack dataset

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

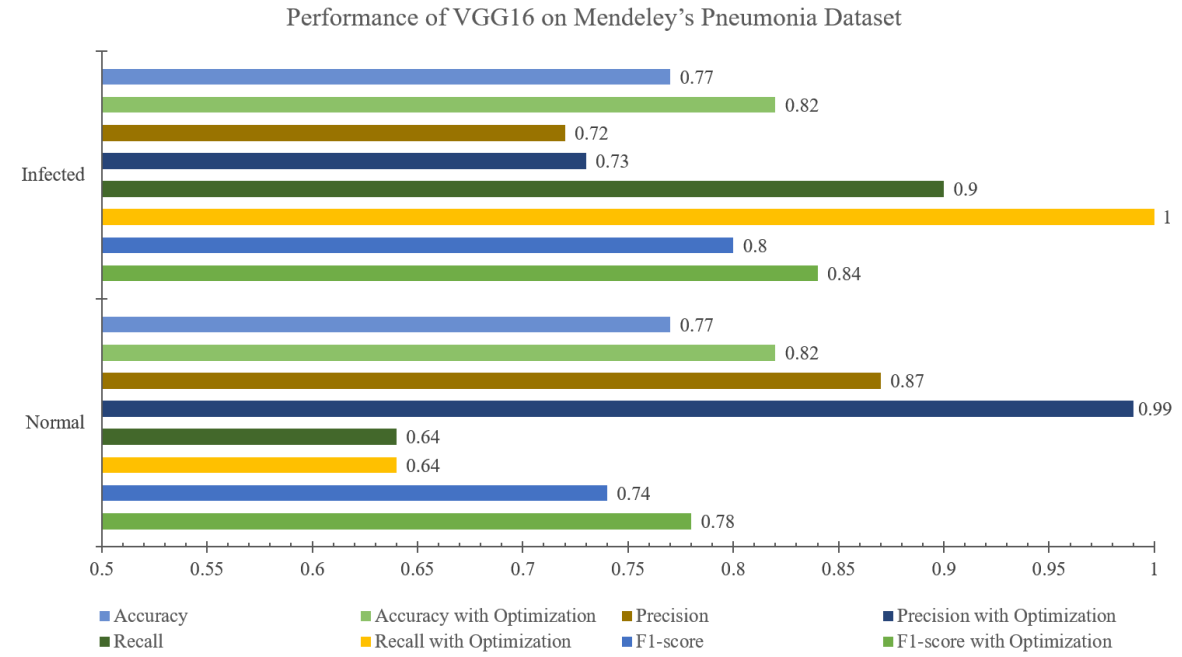

Fig. 9 Performance metrics of VGG16 on Mendeley's pneumonia dataset

The performance characteristics of the optimization approach for VGG16 on Mendeley's pneumonia dataset are depicted in Fig. 9 Performance metrics of VGG16 on Mendeley's pneumonia dataset. Without the optimization, the precision for the infected instances was 0.72, whereas it was 0.87 for the normal cases. The overall precision averaged out to be 0.79. After applying our proposed optimization, the precision for the infected cases went up to 0.73, while the precision for the normal cases went up to 0.99. Without optimization, the recall value came up at 0.90 for infected patients, 0.64 for normal cases, and 0.77 overall, with an average of 0.77. Following the implementation of optimization, the recall value for infected patients was 1.00, while the recall value for normal cases was at 0.64. The overall average was 0.82, which is 0.05 points higher than the previously used methods.

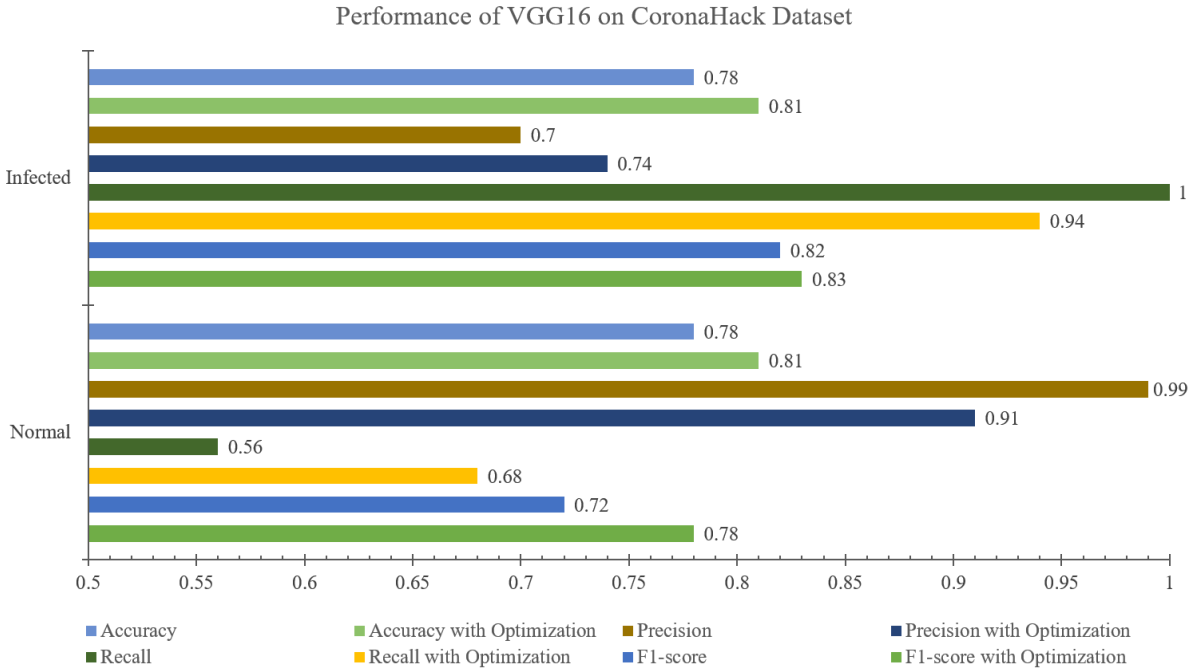

Fig. 10 Performance metrics of VGG16 on CoronaHack dataset

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.021 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 11, Issue 7, July 2023

DOI: 10.17148/IJIREEICE.2023.11718

The F1-score, which is a harmonic mean of the combination of accuracy and recall, was 0.80 for cases that were infected and 0.74 for those that were normal. The F1 score was 0.84 for instances that were infected, while normal cases had a score of 0.78 points. After optimization, the F1 score grew by 0.04 points, bringing the total to 0.81. The average F1 score for the current approaches was 0.77 points. The accuracy indicated a 0.05 point increase after the modification. Before the optimization technique was implemented, the value was at 0.77, but after it was implemented, it climbed to 0.82 points. A similar kind of results was obtained when we did the experiment on the CoronaHack dataset as depicted in Fig. 9 Performance metrics of VGG16 on Mendeley's pneumonia dataset.

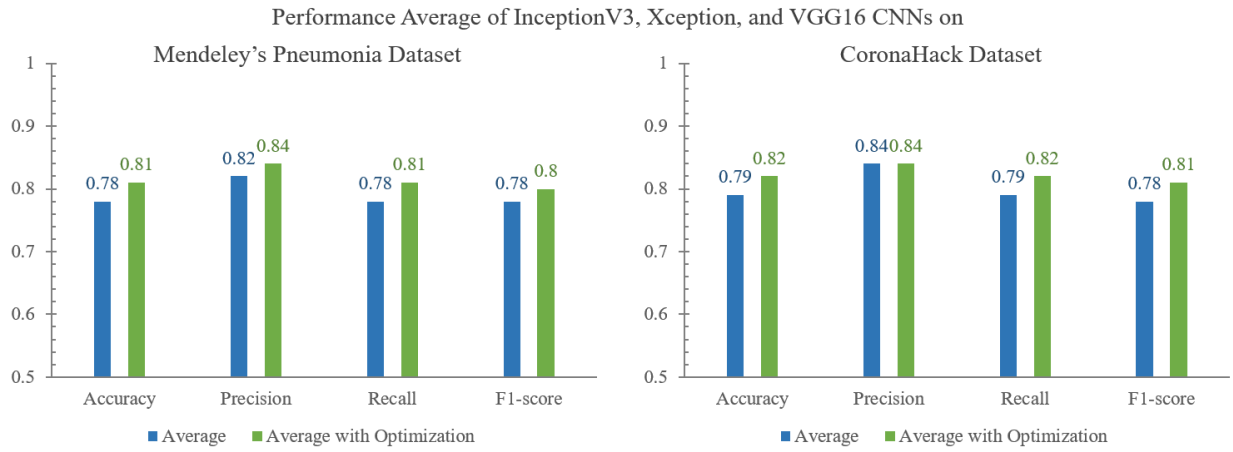

Fig. 11 Average performance of Inception v3, Xception, and VGG16 CNNs on Mendeley's pneumonia dataset and CoronaHack dataset

The average performance of Inception v3, Xception, and VGG16 on Mendeley's pneumonia dataset showed a 0.03 point improvement in accuracy and recall. The precision and F1-score showed an improvement of 0.02 points each. When experimented on the CoronaHack dataset, on average, the Inception v3, Xception, and VGG16 showed an improvement of 0.03 points in accuracy, precision, and F1-score. The recall score grew by 0.03 points. The results are depicted in Fig. 11 Average performance of Inception v3, Xception, and VGG16 CNNs on Mendeley's pneumonia dataset and CoronaHack dataset.

The above results shows promising improvements over the existing methods.

VII. CONCLUSION

The development of reliable techniques for detecting and identifying features in images is a pressing area of research right now. In this research, we improved the performance of a deep neural network with the aid of the efficient data cleaning and optimization techniques. We believe that this will benefit the business by allowing for more accurate predictions. Medical imaging, among other fields, relies heavily on technology with high precision and minimal loss. In this study, we demonstrate how our proposed framework may be tested experimentally and lead to better outcomes. Accuracy, which was previously at 0.78, is now at 0.81 points. When compared to the status quo, this represents an improvement. Therefore, we arrive at the conclusion that better results can be achieved when employing deep neural networks for image processing if the data is cleaned and optimized effectively.

REFERENCES

[1] Fu Y., Zhang H., Morris E. D., Carri K., Hurst G., Pai S., Traverso A., Wee L., Hadzic I., Lonne P. I., Shen C., Liu T., Yang X., "Artificial intelligence in radiation therapy," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 6, p. 2, 2022.

[2] Shi F., Wang J., Shi J., Wu Z., Wang Q., Tang Z., He K., Shi Y., Shen D., "Review of artificial intelligence techniques in imaging data acquisition, segmentation, and diagnosis for COVID-19," *IEEE Reviews in Biomedical Engineering*, vol. 14, 2021.

[3] Alhasan M., Hasaneen M., "Digital imaging, technologies and artificial intelligence applications during COVID-19 pandemic," *Computerized Medical Imaging and Graphics*, vol. 91, p. 101933, 2021.

DOI: 10.17148/IJIREEICE.2023.11718

[4] Canayaz M., "MH-COVIDNet: Diagnosis of COVID-19 using deep neural networks and meta-heuristic-based feature selection on X-ray images," *Biomedical Signal Processing and Control*, vol. 64, p. 102257, 2021.

[5] Gupta M., Kumar N., Gupta N., Zagula A., "Fusion of multi-modality biomedical images using deep neural networks," *Soft Computing*, vol. 26, pp. 8025-8036, 2022.

[6] Samek W., Montavon G., Lapuschkin S., Anders C. J., Muller K. R., "Explaining deep neural networks and beyond: a review of methods and applications," *IEEE Proceedings*, vol. 109, no. 3, 2021.

[7] Ozturk T., Talo M., Yildirim E. A., Baloglu U. B., Yildirim O., Acharya U. R., "Automated detection of COVID-19 cases using deep neural networks with X-ray images," *Computers in Biology and Medicine*, vol. 121, p. 103792, 2020.

[8] Sun Y., Xue B., Zhang M., Yen G. G., "Evolving deep convolutional neural networks for image classification," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 2, 2020.

[9] Sharmila V. J., Jemi F. D., "Deep learning algorithm for COVID-19 classification using chest X-ray images," *Computational and Mathematical Methods in Medicine*, vol. 2021, p. 9269173, 2021.

[10] Heidari M., Mirniaharikandehei S., Khuzani A. Z., Danala G., Qiu Y., Zheng B., "Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms," *International Journal of Medical Informatics*, vol. 144, p. 104284, 2020.

[11] Sirazitdinov I., Kholiavchenko M., Mustafaev T., Yixuan Y., Kuleev R., & Ibragimov B., "Deep neural network ensemble for pneumonia localization from a large-scale chest X-ray database," *Computers and Electrical Engineering*, vol. 78, pp. 388-399, 2019.

[12] Meng H., Gao Y., Yang X., Wang K., & Tian J., "K-nearest neighbor based locally connected network for fast morphological reconstruction in fluorescence molecular tomography," *IEEE Transactions on Medical Imaging*, vol. 39, no. 10, 2020.

[13] Li X., Shen L., Xie X., Huang S., Zhien X., Hong X., Yu J., "Multi-resolution convolutional networks for chest X-ray radiograph based lung nodule detection," *Artificial Intelligence in Medicine*, vol. 103, p. 101744, 2020.

[14] Wu H., Xie P., Zhang H., Li D., & Cheng M., "Predict pneumonia with chest X-ray images based on convolutional deep neural learning networks," *Journal of Intelligent & Fuzzy Systems*, vol. 39, pp. 2893-2907, 2020.

[15] Chen C., Zhou K., Zha M., & others, "An effective deep neural network for lung lesions segmentation from COVID-19 CT images," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 9, 2021.

[16] Zhang M., Li H., Pan S., Lyu J., Ling S., & Su S., "Convolutional neural networks-based lung nodule classification: a surrogate-assisted evolutionary algorithm for hyperparameter optimization," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 5, 2021.

[17] Dixit A., Mani A., & Bansal R., "CoV2-Detect-Net: Design of COVID-19 prediction model based on hybrid DE-PSO with SVM using chest X-ray images," *Information Sciences*, vol. 571, pp. 676-692, 2021.

[18] Vinod D.N., Jeyavadhanam B.R., Zungeru A.M., & Prabaharan S.R.S., "Fully automated unified prognosis of COVID-19 chest X-ray/CT scan images using deep Covix-Net model," *Computers in Biology and Medicine*, vol. 136, p. 104729, 2021.

[19] Sharifrazi D., Alizadehsani R., Roshanzamir M., & others, "Fusion of convolution neural network, support vector machine and sobel filter for accurate detection of COVID-19 patients using X-ray images," *Biomedical Signal Processing and Control*, vol. 68, p. 102622, 2021.

[20] Karnati M., Seal A., Sahu G., Yazidi A., & Krejcar O., "A novel multi-scale based deep convolutional neural network for detecting COVID-19 from X-rays," *Applied Soft Computing*, vol. 125, p. 109109, 2022.

[21] Halder A., Chatterjee S., & Dey D., "Adaptive morphology aided 2-pathway convolutional neural network for lung nodule classification," *Biomedical Signal Processing and Control*, vol. 72, p. 103347, 2022.

[22] Mahmood S.A., & Ahmed H.A., "An improved CNN based architecture for automatic lung nodule classification," *Medical & Biological Engineering & Computing*, vol. 60, pp. 1977-1986, 2022.

[23] Maity A., Nair T.R., Mehta S., & Prakasam P., "Automatic lung parenchyma segmentation using a deep convolutional neural network from chest X-rays," *Biomedical Signal Processing and Control*, vol. 73, p. 103398, 2022.

[24] Aslan M.F., Sabanci K., Durdu A., & Unlersen M.F., "COVID-19 diagnosis using state-of-the-art CNN architecture features and Bayesian optimization," *Computers in Biology and Medicine*, vol. 142, p. 105244, 2022.

[25] Jalali S.M.J., Ahmadian M., Ahmadian S., Hedjam R., Khosravi A., & Nahavandi S., "X-ray image based COVID-19 detection using evolutionary deep learning approach," *Expert Systems with Applications*, vol. 201, p. 116942, 2022.

DOI: 10.17148/IJIREEICE.2023.11718

[26] Min L., Ming Y., Minghu W., Juan W., Yu H., "Breast pathological image classification based on VGG16 feature concatenation," *Journal of Shanghai Jiaotong University (Science)*, vol. 27, no. 4, pp. 473-484, 2022.

[27] Chen Y., Lin Y., Xu X., Ding J., Li C., Zeng Y., Liu W., Xie W., Huang J., "Classification of lungs infected COVID-19 images based on inception-ResNet," *Computer Methods and Programs in Biomedicine*, vol. 225, p. 107053, 2022.

[28] Polat O., "Detection of COVID-19 from chest ct images using Xception architecture a deep transfer learning based approach," *Sakarya University Journal of Science*, vol. 25, no. 3, pp. 800-810, 2021.

[29] Chen B., Chen L., Chen Y., "Efficient ant colony optimization for image feature selection," *Signal Processing*, vol. 93, pp. 1566-1576, 2013.

[30] Zhang H., Nguyen H., Bui X. N., Thoi T., N., Bui T. T., Nguyen N., Vu D. A., Mahesh V., Moayedi H., "Developing a novel artificial intelligence model to estimate the capital cost of mining projects using deep neural network-based ant colony optimization algorithm," *Resources Policy*, vol. 66, p. 101604, 2020.

[31] Zhao D., Liu L., Yu F., Heidari A. A., Wang M., Oliva D., Muhammad K., Chen H., "Ant colony optimization with horizontal and vertical crossover search: Fundamental visions for multi-threshold image segmentation," *Expert Systems with Applications*, vol. 167, p. 114122, 2021.

[32] Berger B., Waterman M. S., Yu Y. W., "Levenshtein distance, sequence comparison and biological database search," *IEEE Transactions on Information Theory*, vol. 67, no. 6, 2021.

[33] Qi A., Zhang D., Yu F., Heidari A. A., Wu Z., Cai Z., Alenezi F., Mansour R. F., Chen H., Chen M., "Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation," *Computers in Biology and Medicine*, vol. 148, p. 105810, 2022.

[34] Paniri M., Dowlatshahi M. B., Pour H. N., "MLACO: A multi-label feature selection algorithm based on ant colony optimization," *Knowledge-Based Systems*, vol. 192, p. 105285, 2020.

[35] Liu L., Zhao D., Yu F., Heidari A. A., Li C., Ouyang J., Chen H., Mafarja M., Turabieh H., Pan J., "Ant colony optimization with Cauchy and greedy Levy mutations for multilevel COVID 19 X-ray image segmentation," *Computers in Biology and Medicine*, vol. 136, p. 104609, 2021.

[36] Kermany D., Zhang K., Goldbaum M., "Labeled optical coherence tomography (OCT) and chest X-ray images for classification," Mendeley Data, 2018.

[37] Praveen G., "Coronahack - Chest X-ray-dataset," Kaggle, 2020.

[38] Ye L. Y., Miao X. Y., Cai W. S., Xu W. J., "Medical image diagnosis of prostate tumor based on PSP-Net + VGG16 deep learning network," *Computer Methods and Programs in Biomedicine*, vol. 221, p. 106770, 2022.

[39] Kong L., Cheng J., "Classification and detection of COVID-19 X-ray images based on DenseNet and VGG16 feature fusion," *Biomedical Signal Processing and Control*, vol. 77, p. 103772, 2022.

[40] Dong N., Zhao L., Wu C. H., Chang J. F., "Inception v3 based cervical cell classification combined with artificially extracted features," *Applied Soft Computing Journal*, vol. 93, p. 106311, 2020.

[41] Ganguly S., Ganguly A., Mohiuddin S. K., Malakar S., Sarkar R., "ViXNet: Vision transformer with Xception network for deepfakes based video and image forgery detection," *Expert Systems with Applications*, vol. 210, p. 118423, 2022.

BIOGRAPHY

Mr. **Suresh Mohan** received his M.Sc. (Computer Technology) degree from Anna University, Chennai, India. He is currently working as a software engineer in the IT industry. His research interests include data science and machine learning techniques.