Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
A three-stage ensemble boosted convolutional neural network for classification and analysis of COVID-19 chest x-ray images

S. Kalaivani, K. Seetharaman
Dept. of Computer and Information Science, Annamalai University, Annamalai Nagar, 608 002, Tamil Nadu, India

A R T I C L E   I N F O

Keywords:
Three-stage ensemble boosted CNN
CXR images
COVID-19
Machine Learning
ResUnet

A B S T R A C T

For the identification and classification of COVID-19, this research presents a three-stage ensemble boosted convolutional neural network model. A conventional segmentation model (ResUNet) is used to increase the model's performance in the initial step of processing the CXR datasets. In the second step, the CNN is used to extract the features from the pictures in the training dataset using machine learning techniques. Using machine learning (ML) techniques, the retrieved characteristics are then combined by voting in the third stage. There are 5178 aberrant CXR photos and 4310 normal CXR images used in this investigation. Models like CNN and ML can't compete with the suggested model. 99.35% of the model's measurements are accurate and precise, and 98% of its recall and F1-score are perfect. It is argued that the suggested model provides a rigorous and trustworthy evaluation of clinical decision-making in the setting of a public health crisis.

1. Introduction

Severe Respiratory Syndrome Corona infection 2 (SARS-CoV-2) was discovered in Wuhan, China, in January 2020, and the World Health Organization (WHO) has labelled it a pandemic. (WHO, 2020). In March 2020, the World Health Organization (WHO) declared this pandemic a Public Health Emergency of International Concern (Bhagavathula, Aldhaleei, Rahmani, Mahabadi & Bandari, 2020). To far, more than 36 million individuals have been infected with the extremely infectious COVID-19 virus, which has resulted in over a million fatalities (Dong, Du & Gardner, 2020). As a result of a shortage of centres, poor health care, and ineffective analytical methods, all countries, developed, developing, and undeveloped alike are battling the epidemic. High fever, cough, and other severe respiratory symptoms are telltale indicators of COVID-19 infection (Zu et al., 2020). Because of this, it is critical to diagnose the infection in order to prevent its spread. RT-PCR, Computed Tomography (CT), and Chest X-Ray (CXR) are the three primary screening methods employed in COVID-19 identification (Wang et al., 2020). (CXR). Using the blockchain, Manoj et al. (2020) have developed a system that maintains track of every individual's COVID data in the country while keeping track of whether an individual has been tested positive, is suspected of having the virus, or is otherwise healthy. Reverse transcription-PCR (RT-PCR) is one of the most often used assays for COVID-19 detection, which uses PCR to amplify DNA for analysis. COVID-19 can be detected since the infection is exclusively RNA-based (Corman et al., 2020). COVID-19 detection takes anything from a few hours to two days, and the test kit isn’t widely distributed. Consequently, healthcare systems and medical professionals face substantial challenges in dealing with the pandemic due to the lack of reliability of this test (Lauer et al., 2020).

Instead of using RT-PCR and quarantine to curb this pandemic, researchers are developing other services. Medical diagnosis of COVID-19 is based on radiological image analysis (Corum & Zimmer, 2020; Islam et al., 2022). In this case, CT scans or chest X-rays play a vital role. X-ray or CT pictures of infected upper body with COVID-19 show ground-glass opacity or unclear darker spots, which can help researchers discover COVID-19 (Xia et al., 2020). A pandemic situation necessitates a cutting-edge, easily available, and fast-assessing technology for radiologists to use in the early diagnosis of COVID-19. COVID 19 may spread quickly because of the time-consuming nature of the CXR image review procedure. Making industrialised and improved automated systems for recognising COVID-19 contaminated photos is therefore required.

Deep learning has shown a new path in the healthcare system (Rastgarpour & Shabnezhadeh, 2011). The healthcare system has revealed great innovation for automatic disease identification, such as the detection of chest diseases (Hwang et al., 2019), the detection of cancer cells (Ciresan, Giusti, Gambardella & Schmidhuber, 2013), the recognition of skin cancer (Seetharaman, 2018, 2019), tumour detection (Mohsen, El-Dahshan, El-Horbaty & Salem, 2018) and genomic sequence evaluation, with the contribution of a deep neural network (DNN).

* Corresponding author.
E-mail address: kseethaddeau@gmail.com (K. Seetharaman).
(Quang, Xie & Dan, 2016), Kieu, Tran, Le, Le and Nguyen (2018) proposed a deep learning model that can identify CXR pictures with an exceptional density. It was possible to detect normal or abnormal densities in upper body X-ray pictures using three CNN models (CNN-128F, CNN-64 L, and CNN-64R). In this dataset, 400 CXR pictures have been used, with a 3:1 ratio of training to testing. For normal photographs, we used the code 0; for aberrant images, we used the code 1. Extensive testing reveals a precision of 96%. They have created numerous deep semantic network versions for attribute removal from COVID-19-infected breast X-ray pictures, ensuring remarkable precision, degree of sensitivity, and uniqueness. This is an ongoing effort by many researchers. In the absence of COVID-19 photos, several of them used a tiny dataset to train their model. In order to minimise over-fitting, Wang et al. (2021) suggested a deep rank-based average pooling network. For example, strided convolution, l2-norm pooling and average pools, as well as max pools. If you combine the CCT and CXR pictures into one, you may create a multiple-input deep convolutional attention network (MIDCAN), which processes both images concurrently, according to Zhang, Zhang, Zhang and Wang (2021). As a result, the current approaches might use some refinement in the form of more effective training and assessment. Our approach to detecting COVID-19 CXR pictures employs a three-stage ensemble Boosted Convolutional Neural Network. The full image does not need to be examined in medical image processing, though. Consequently, CNN has to concentrate on its immediate surroundings while also improving its overall output. Pre-processed and segmented CXR pictures are used in the benchmark dataset. After that, in the feature extraction phase, the CNN model extracts deep features from CXR pictures in order to identify those afflicted with COVID-19 infection. Finally, a collection of classifiers is assembled for classification. Classification may be done using any of the ML models. Although a model that performs exceptionally well on one piece of data does not ensure that it will perform as well on all future sets of data. This means that different models will provide different results when applied to different datasets. As a result, in this study, four ML classifiers were integrated to form an ensemble of classifiers, which ensures superior results for datasets of varying sizes and resolutions.

The following is a summary of the rest of the paper. Section 2 explores the use of deep learning to detect COVID-19 in CT scans and CXR radiography pictures. The dataset and the proposed three-stage ensemble Boosted Convolutional Neural Network model and its structure are described in detail in Section 3. The suggested method’s performance is evaluated in Section 4 using performance metrics and a comparative comparison with currently in use systems. Section 5 concludes with the final paragraph.

2. Related works

For a number of years now, deep learning approaches have demonstrated their capacity to solve a wide range of problems, particularly in computer vision. Non-invasive clinical adjuncts such as the chest X-ray (CXR) play an important role in the first assessment of various lung disorders (Chandra & Verma, 2020, 2020a; Chandra, Verma, Jain & BNetam, 2020; Ke et al., 2019). Using clinical imaging system deep feature extraction techniques, the advances in computing capability made possible by the availability of large labelled datasets have been widely publicised (Kim et al., 2019). CXR pictures, one of the most widely used imaging procedures in clinical practise, have been utilised to examine these techniques’ usefulness in detecting and assessing cardiac, thoracic, and pulmonary issues (Monsi, Poon & Chung, 2020; Tang et al., 2020). Because of this, Anthimopoulos, Christodoulidis, Ebner, Christ and Mougiakakou (2016) developed a deep CNN for classifying lung image patches into seven distinct categories based on six distinct patterns of interstitial lung disease and healthy, balanced cells. The emphysema section of replacement CXR pictures may be quantified using a method based on the CNN provided by Campo, Pascau and Estépar (2018). In order to identify pneumonia in CXR pictures, Jaiswal et al. (2019) developed a deep semantic network that uses global and local information. In Passa, Golkov, Pfeiffer, Cremers and Pfeiffer (2019), a deep network architecture for consumption diagnostics has been shown. Others in this field use transfer discovery algorithms to perform a reliable understanding step over a restricted dataset utilising characteristics extracted from other large datasets in the same domain name.

As a result, the first technique may classify patients into two groups: those with COVID-19 and those with other respiratory issues that are equivalent. As an alternative to COVID-19, a second approach uses the patient’s CXR pictures to identify any other present or normal respiratory system issues. The third strategy allows for categorization between the clinical categories at the same time, which is very useful. When these methodologies are used, it is more easier for the researchers to detect COVID-19, abnormal and usual scenarios.

However, despite the poor quality of the CXR pictures caused by the portable equipment, the offered methods provided global accuracy worth 79% to 90%. Help in clinical decision-making by enabling the reliable analysis of portable radiography. Using the discovery of five different semantic network designs, Chouhan et al. (2020) offer a new ensemble technique based on the transfer discovery. For COVID-19 screening on CXR images, Zhang et al. (2020) suggest a deep anomaly finding design. Using a deep learning version, Ozurk et al. (2020) advocated emphasising critical regions in CXR pictures to aid professionals in the early detection of COVID-19 disease. It is proposed that CXR pictures should be classified into usual courses, pneumonia, TB, and COVID-19 (Shelke et al., 2020). COVID-19 is also divided into three categories: mild, medium, and severe. In order to find COVID-19 in breast X-ray pictures, Voo et al. (2020) propose a deep learning-based decision-tree classifier. Initially, the photos are classified as normal by a first decision tree, which is followed by a second tree that detects images with irregularities indicative of TB, and a third tree that identifies scientific results associated with COVID-19. In order to detect incubation or death in patients with COVID-19, Li et al. (2020) used a Convolutional Siamese neural network using CXR pictures. Moura et al. (2020) have developed a deep learning method based on CNN that uses data augmentation and regularisation approaches to arrange the unbalanced COVID-19 information. Pre-processing tests were conducted in three distinct ways. In order to ensure the safety of the system, a thorough investigation is carried out into a variety of issues that might affect its performance. It has a 91.5 percent accuracy rate and an 87.4 percent recall rate. In Wang et al. (2020), chest radiography pictures were advised to use the COVID-Net architecture. An open data collection of 13,975 CXR pictures, dubbed COVIDs, was used to create the network. The COVID-19 class has just 358 samples derived from 266 people. The obtained precision was 93.3%. For the COVID-19 medical diagnosis using CXR pictures, Singh and Singh used an automated Wavelets-Based Depth-wise Convolution Network (2021).

The neural network used to evaluate the CXR pictures is enhanced using this technique’s depth-wise convolution neural network. Multi-resolution analysis in the network is integrated using wavelet decay. The input pictures’ frequency subbands are supplied into the network to determine the current state. The network classifies the input picture into categories like “normal,” “viral pneumonia,” and “COVID-19,” amongst others. Grad-CAM is used to see the model’s output in order to aid in diagnosis. There are measures for analysing performance and verifying that the method accurately identifies sickness such as accuracy, sensitivity, and factor 1. An iteratively pruned deep learning design ensemble was used to detect pulmonary symptoms of COVID-19 using CXR, according to Rajaraman et al. (2020). Open CXR datasets are used to train and evaluate custom convolutional neural networks and ImageNet pre-trained versions for recognising modality-specific function descriptions at the individual patient level. For the linked job of classifying CXRs as either normal, bacterial pneumonia, or anomalies of COVID-19-viral illness, the learnt intelligence is relocated and calibrated. Integration of forecasts from the best-performing trimmed models is achieved through a variety of different methods. Better projections were achieved by the
application of modality-specific knowledge transfer, recurrent version reduction, and ensemble learning. We anticipate that using breast radiographs, the suggested approach may be quickly implemented for COVID-19 screening.

3. Methodology

CXR pictures from a variety of sources were compiled to test the feasibility of the suggested strategy. For the sake of generalisation and avoiding overfitting, the CXR pictures were scaled and normalised, and a dataset of preprocessed images was created. The CXR pictures were then segmented to increase the suggested model’s accuracy. Training, validation, and testing sets were created using the dataset’s photos. Training and validation data are used to train the intended CNN model. To begin, all of the training pictures are used to extract their features, which are then combined into a feature vector using a Fully Connected Layer (FCL).

There are four ML classifiers that get feature vectors from the training CXR pictures. Each ML classifier is given the grid search technique in order to fine-tune the hyper-parameters that make up an ideal ML classifier. All the ML classifiers are eventually linked together to form an ensemble of classifiers. The ensemble classifier is a collection of classifiers that compares the results of classifications made using the same data by a number of different models. Decision Tree, Random Forest, Ada Boost, and SVM receive the most votes in the proposed system. As a result, training data may be more accurately labelled. As a consequence, class labels are decided by a majority vote of a group of classifiers. Fig. 1 depicts the suggested training model’s overall working principle, and accuracy, precision, recall, and F1 score are used to gauge its effectiveness.

3.1. Dataset description

We built a new database including CXR images from a variety of sources, such as Github (Cohen et al., 2020), SIRM (SIRM), TCIA (TCIA), and radiopaedia to improve the proposed model’s classification and generalisation capability for the COVID-19 identification (Saha et al., 2018). Pneumonia cases as well as COVID-19 positives and negatives make up the newly built CXR image database. It’s not clear how many people were infected, but the average age of those afflicted was between 50 and 55 years old. There were 4260 COVID-19-infected photos in the database at the time of study, and 918 images were infected with various illnesses such viral and microbial pneumonia (MERS, SARS, and ARD), as well as 4310 CXR-normal images. All other viral and bacterial pneumonia illnesses are omitted from the proposed detection model for COVID-19. Over the course of the study, 9488 CXR pictures were used for testing. Fig. 2 shows a few examples of the CXR pictures that were used for training, validation or testing, using a 7:2:1 ratio, as shown in Table 1.

3.2. Proposed model

3.2.1. Stage 1: pre-processing and segmentation process

CXR images are segmented using UNet (Ronneberger, Fischer & Brox, 2015) with ResNet backbone (ResNet) for the lung regions in this initial step. The design incorporates a narrowing route for capturing context and symmetric expanding facilitates for precise localisation, both of which are key components. Essentially, the concept is to use succes-

| Dataset | Abnormal | Other diseased | Normal | Total |
|---------|----------|----------------|--------|-------|
| Training | 2982 | 756 | 3017 | 6755 |
| Validation | 852 | 108 | 862 | 1822 |
| Testing | 426 | 54 | 431 | 911 |
| Total | 4310 | 918 | 4310 | 9488 |
sive layers of upsampling operators and pooling operators to improve a standard contract network. In this way, the resolution of the output is improved by these layers. Contracted paths and upsampled results are used to find high-resolution features. In Fig. 3, the findings reveal that the lung masks may be predicted more accurately with the help of the subsequent convolution layer.

3.2.1.1 UNet. Use low-degree details while keeping high-degree semantic information in semantic segmentation to achieve a better result (Ronneberger et al., 2015). It is difficult to train a deep neural network, especially when there are just a few training candidates. Using a pre-trained network after fine-tuning it on the target dataset is one way to address this problem. Using complete data augmentation, like in UNet, is still another option (Ronneberger et al., 2015). U-approach, Net's in addition to data enrichment, we feel helps alleviate the training issue. By replicating low-level features at higher levels, progress is made for information propagation, allowing signals to spread between levels in an easier manner, not only promoting backward propagation during training but also making up reduced level better information for high degree semantic attributes. This is the instinct behind this.. Similarities may be drawn between the residual neural network and this (He et al., 2016). U-performance Net's can be increased by using a repeating device instead of the basic system described in this work.

3.2.1.2. Recurring unit. Going much deeper would improve the performance of a multi-layer neural network, despite interfering with the training and a deterioration issue possibly occur (He et al., 2016). To get rid of these troubles, He et al. (2016) have proposed a residual neural network for training and solve the degradation trouble. The residual semantic network is composed of a series of piled recurring units. Each residual system can be shown as a general type as in Eq. (1) and (2).

\[ y_r = h(x_r) + F(x_r, W_r) \]  \hspace{1cm} (1)

\[ x_{r+1} = f(y_r) \]  \hspace{1cm} (2)

where, \( x_r \) and \( x_{r+1} \) are the input and result of the \( r \)-th residual system, \( F(\cdot) \) is the recurring feature, \( f(\cdot) \) is activation feature, and \( h(\cdot) \) is an identification mapping feature, a regular one \( h(x_r) = x_r \). Fig. 4 shows the distinction between a simple and residual system. There are multiple mixtures of batch normalization (BN), ReLU activation, and convolutional layers in a residual unit. He et al. offer a comprehensive discussion on the effects of various combinations. The ResUnet, additionally utilizes a full pre-activation residual unit to construct the deep recurring U-Net.

3.2.1.3. ResUnet. Adapting the U-Net and the recurring neural network's characteristics, ResUnet creates a semantic segmentation neural

Fig. 2. row 1 - Sample COVID-19 images; row 2 - normal images; row 3 - Pneumonia CXR images.
Fig. 3. Outcomes of UNet Segmentation. row 1: given input images; row 2: Segmented output results of the corresponding images in row 1.

Fig. 4. Block diagram of Neural Networks. (a): Neural units used in U-Net; (b): Residual unit with identity mapping used in the ResUnet.
network. There are two advantages to this fusion: (2) eliminates linkages amongst recurrent devices and between the low and high degrees of the network, which unquestionably aids in information transmission without destruction, the residual system simplifies training the network.

In order to improve semantic segmentation efficiency, it is possible to build a neural network with fewer parameters. ResUnet has a seven-layer architecture, as shown in Fig. 5 for CXR image segmentation of the lungs. Encoding, bridging, and decoding make up the network’s three main components. Encoding and decoding are linked in the second portion of the process. Segmentation at the semantic level is performed as the final step in the process. All three components, including two 3 × 3 convolution blocks and an identification mapping, are constructed using residual systems. BN, ReLU activation, and a convolutional layer are all included in each convolution block. The unit’s input and output are linked together via an identification mapping. In the encoding sequence, there are three remaining units. Instead of employing the pooling process, each unit reduces the future map dimension. The first convolution block uses atride of 2 to shrink the feature map by 50%. The decoding process, on the other hand, generates three repeated units. A concatenation of the feature maps from the corresponding encoding path is performed with the enhanced feature maps from the reduced levels before each unit. The multi-channel feature maps are projected onto the chosen segmentation using an 11 convolution and a sigmoid activation layer after the last degree of decoding. U-Net has 23 layers, whereas our convolutional network has 15.

### 3.2.2. Stage 2: feature extraction process

In the second step, features are extracted from the segmented CXR pictures using a CNN network. Convolutional, pooling, and fully linked are the three main layers of CNN (FCL). Table 2 displays the many types of CNN model layers and provides an explanation for each one. The RGB colour pictures (224 × 2243 × 3) are put into the CNN model to train the proposed model. Three representations () of the input image are available for processing. Kernel (fn) of dimension fn functions as a filter to represent each column and each row of a window in the convolutional layer. As the name suggests, it is a “feature identifier.” Low-level characteristics like edges and contours are added to the layer using these filters. Convolution is performed on a portion of the picture by the filter. The convolution process involves a step-by-step replication and summation of the image’s filter and pixel values. Weights and criteria are two terms used to describe the filter’s values. The version must be trained to learn these weights. More convolutional layers improve the model’s ability to remove pictures’ deep properties, making it possible for this version to detect all of an image’s features. A responsive image is the picture’s sub-image. It begins convolution at the beginning of the picture and continues to do so until the entire image has been covered by the filter. Throughout the pictures, convolution generates a metric or range of values. The operation is expressed as in Eq. (3).

\[
C = I \times F = \sum \sum I(i + f, j + n) \times F(f_h, f_w)
\]

(3)

where, \(I\) is the input and \(F\) is the filter with size \(f_h\) and \(f_w\). The operation is represented by the operator \(\times\).

An additional parameter, stride, identifies the change in filter amount. All convolutional layers have their stride set to 1 for the version. With each step up in stride, the amount of input that can be measured in terms of height and width decreases. An excessively high stride value might lead to difficulties, such as the response size exceeding input amount and reducing measurements. There are a number of ways to deal with these issues, such as zero padding (sometimes referred to as "the same padding"), which is a method of padding an input with no padding around its borders and keeping its output amount measurement constant if the stride length is 1, is identified by Eq. (4).

\[
\text{Zero Padding} = \frac{j - 1}{2}
\]

(4)

where, height or size of the filter is represented by \(j\); both height and width are the same size. In stage two, valid extra padding was made instead of absolutely no padding, so the output image is not the same as the input, and its size is reduced after the convolution.

Numerous filters in the convolutional layer have been made use of extracting several functions. In the first layer, 32 filters were used. The variety of filters increased gradually in the successive layers, from 32 to 128, 128 to 512, and so on. The resulting quantity is called activation map or feature map. In Table 3, all the layers result in shape is shown, and the size of result volume is determined by Eqs. (5), (6), and (7).

\[
O_h = \frac{I_h - f_h + 2P}{S} + 1
\]

(5)

\[
O_w = \frac{I_w - f_w + 2P}{S} + 1
\]

(6)

\[
O = N_f
\]

(7)

Where \(I_h\) means height, \(I_w\) represents width, \(f_h\) is the filter height, \(I_w\) is the filter width, \(S\) denotes the stride size, \(P\) denotes padding, and \(N_f\) is the number of filters.

For the 1-st convolutional layer, \(I_h = 224\), \(I_w = 224\), \(f_h = 3\), \(f_w = 3\), \(S = 1\), \(P = 0\) and \(N_f = 32\).

| Layer          | Filter Size | Pool Size | Stride | Padding | No. of Filters | Dropout Threshold | Activation |
|----------------|-------------|-----------|--------|---------|----------------|-------------------|------------|
| Conv2D        | 3 × 3       | –         | 1      | Valid   | 32             | –                 | Relu       |
| Conv2D        | 3 × 3       | –         | 1      | Valid   | 128            | –                 | Relu       |
| MaxPooling2D  | –           | 2 × 2     | 2      | –       | –              | –                 | –          |
| DropOut       | –           | –         | –      | –       | 0.25           | –                 | –          |
| Conv2D        | 3 × 3       | –         | 1      | Valid   | 64             | –                 | Relu       |
| MaxPooling2D  | –           | 2 × 2     | 2      | –       | –              | –                 | –          |
| DropOut       | –           | –         | –      | –       | 0.25           | –                 | –          |
| Conv2D        | 3 × 3       | –         | 1      | Valid   | 128            | –                 | Relu       |
| MaxPooling2D  | –           | 2 × 2     | 2      | –       | –              | –                 | –          |
| DropOut       | –           | –         | –      | –       | 0.25           | –                 | –          |
| Conv2D        | 3 × 3       | –         | 1      | Valid   | 512            | –                 | Relu       |
| MaxPooling2D  | –           | 2 × 2     | 2      | –       | –              | –                 | –          |
| DropOut       | –           | –         | –      | –       | 0.25           | –                 | –          |
| Flatten       | –           | –         | –      | –       | –              | –                 | –          |
| FCL           | –           | –         | –      | –       | 64             | –                 | Relu       |
| DropOut       | –           | –         | –      | –       | 0.25           | –                 | –          |
| FCL           | –           | –         | –      | –       | 2              | –                 | Sigmoid    |

---

S. Kalaivani and K. Seetharaman  
International Journal of Cognitive Computing in Engineering 3 (2022) 35–45
Fig. 5. The architecture of the RestUnet.
Table 3

Summary of the CNN Model.

| No. of Layers | Layer                      | Output Shape          | parameter |
|---------------|----------------------------|-----------------------|-----------|
| 1             | conv2d_6 (Conv2D)          | [222,222,32]          | 896       |
| 2             | conv2d_7 (Conv2D)          | [220,220,128]         | 36,992    |
| 3             | max_pooling2d_3 (MaxPooling2D) | [110,110,128]     | 0         |
| 4             | dropout_6 (Dropout)        | [110,110,128]         | 0         |
| 5             | conv2d_8 (Conv2D)          | [108,108,64]          | 73,792    |
| 6             | max_pooling2d_6 (MaxPooling2D) | [54,54,64]        | 0         |
| 7             | dropout_7 (Dropout)        | [54,54,64]            | 0         |
| 8             | conv2d_9 (Conv2D)          | [52,52,128]           | 73,856    |
| 9             | max_pooling2d_7(MaxPooling2D) | [26,26,128]         | 0         |
| 10            | dropout_8 (Dropout)        | [26,26,128]           | 0         |
| 11            | conv2d_10 (Conv2D)         | [24,24,512]           | 590,336   |
| 12            | max_pooling2d_8 (MaxPooling2D) | [12,12,512]       | 0         |
| 13            | dropout_9 (Dropout)        | [12,12,512]           | 0         |
| 14            | conv2d_11 (Conv2D)         | [10,10,512]           | 2,359,808 |
| 15            | max_pooling2d_9 (MaxPooling2D) | [5,5512]        | 0         |
| 16            | dropout_10 (Dropout)       | [5,5512]              | 0         |
| 17            | flatten_1 (Flatten)        | [12,800]              | 0         |
| 18            | dense_2 (Dense)            | [64]                  | 819,264   |
| 19            | dropout_11 (Dropout)       | [64]                  | 0         |
| 20            | dense_3 (Dense)            | [2]                   | 65        |

From Eqs. (5), (6), and (7), the following values can be obtained.

\[ O_h = \frac{224 - 3 + 0}{1} = 222 \]

\[ O_w = \frac{224 - 3 + 0}{1} = 222 \]

\[ O_s = 32 \]

Non-linear activation is used on convolution results. The Convolutional layer has carried out linear computation (element-wise multiplication and also summation). Hence, nonlinearity is presented on the linear procedure with this activation. Activation of ReLU (rectified linear unit) is applied on convolution output. The feature for the ReLU procedure and it is expressed in (8).

\[ \text{ReLU}(X) = \max(0, X) \] (8)

In this case, X represents the output of the convolution procedure. ReLU reduces the likelihood of any negative consequences to nil. ReLU is used in this suggested version because it enhances the nonlinearity of the design and assists in making the calculation time significantly faster without affecting model fidelity. When the lower layers are learnt more slowly, the vanishing gradient problem is reduced. Following two convolution layers, the max-pooling layer is employed. Reduces input’s spatial dimension (both in height and breadth) through this layer. The layer’s stride is 2 in the preferred design, which uses a 2 \( \times \) 2 filter. Convolution around the input volume produces the best possible receptive area output. The relative placement of a feature in relation to other features is more important than the precise location of a feature when using this layer of monitoring. Reduces computational costs by preventing over-fitting and reducing the number of weights. The dropout layer is then used, and it arbitrarily drops out some activation by setting it to zero. Although certain activations remain, this layer ensures that the model is able to predict the true class label of a picture. As a result, fitting the model to the dataset is unnecessary. In order to avoid over-fitting, the dropout layer is used. Previously, dropout layers employed a criterion of 0.25 as their threshold. The FCL uses a flatten layer to transform the 2D function map into a 1D feature vector. Images’ finer details have been accentuated. In order to classify COVID-19, FCL uses 1D long features and then transfers the result activation to a second FCL. Softmax activation is predicted to occur in layers above and below the result layer, hence the model predicts classes between COVID-19 and “normal.”

Table 4

Tuned hyper-parameters of ML classifiers.

| ML classifiers | Hyper-parameters          |
|----------------|---------------------------|
| Random Forest  | Bootstrap: True (method for sampling - with or without replacement) |
|                | max depth: 100 (max number of levels in the decision tree) |
|                | max features: 2 (maximum features for splitting) |
| Support Vector Machine | C: 1 (Regularization parameter) |
|                | Kernel: 'rbf' (Specifies the kernel type: 'linear', 'poly', 'rbf', 'sigmoid') |
| Decision Tree  | max-leaf nodes: 10 min samples split: 4 Criterion: entropy max depth: 6 |
| AdaBoost       | No. of estimators: 250 |

3.2.3. Stage 3: ensemble of classifier

Picked up were features created by the initial FCL of CNN photos. Table 3 shows that the first FCL had 64 nerve cells and was called ‘dense 2.’ As a result, the dimension of the Feature vectors is 164. For each training image, 64 characteristics were extracted from the depths of the image. To train the ML classifiers, 3220 photos are used in the training collection, and the new input is 3220 \( \times \) 64. Table 4 shows the hyper-parameters of the ML classifier that were tweaked using the input data. The ML classifiers’ hyper-parameters were modified to use in the grid search technique. Five-fold cross-validation was used on the various ML models to predict the best classifiers. All the ML classifiers are linked together to form an ensemble of classifiers. To arrive at the final class label, a majority of the classifiers in the ensemble tallied their results from four separate machine learning models. It improved efficiency and performed better than any one classifier could have done on its own. An extremely large number of other versions’ votes were utilised to generate classifier selections during assembly. The Testing stage of the ML classifiers is shown in Fig. 6.

4. Experimental analysis

The proposed three-stage ensemble boosted CNN extracts features from CXR images using CNN and four different ML classifiers ensemble to identify COVID-19. The CNN model was trained, and a grid search technique was employed to tune the hyper-parameters of the ML classifiers. The efficiency of the designed CNN, ML classifiers and an ensemble of classifiers were examined for each class label in terms of accuracy, recall, precision, and F1-score that have described in Eqs. (9), (10), (11), and (12), respectively.

\[ \text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \] (9)

\[ \text{Recall} = \frac{TP}{TP + FN} \times 100 \] (10)

\[ \text{Precision} = \frac{TP}{TP + FP} \times 100 \] (11)

\[ F1 - score = 2 \times \frac{\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \times 100 \] (12)

Where, TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively.

4.1. Experimental setup

The proposed method was implemented using Python 3.7x with additional libraries such as Pandas, Tensor Flow, and Keras for training.
the CXR COVID-19 dataset. Windows 10 Operating System powered the System with configuration, Core i5 and 11-th generation.

4.2. Results and analysis

Table 1 shows the use of the picture datasets given in Section 3.1 for training, validation, and testing. A total of 9488 photos were used in the research. There were 4261 pictures with COVID infection; 2982, 852, 426 were used for training, validation, and testing; 918 756, 108, and 54 were used for training, validation, and testing of other sick photos. A total of 4310 normal photos were considered for training, validation, and testing, with 3017, 862, and 431 images being used for each step. COVID-19 was identified using a variety of performance indicators, including accuracy, precision and recall, as well as the F1 Score. Precision, recall, accuracy, and F1 Score for several classifiers applied to each picture category, such as COVID-19 and Normal, were tested for the proposed technique and the results are shown in Table 5. Classifiers for Ensemble Boosted CNN have been evaluated and are shown in Table 6.

While the F1 score is 98.90 percent for COVID infected pictures, the CNN approach obtains an accuracy rate of 97.83 percent for precision, 97.83 percent for recall, and 100% for Accuracy, Precision, Recall and F1-score for COVID-infected images. If you use the RF model to analyse COVID photos, it has a 95.66% accuracy rate; it also has a 95.65% recall.

| Models   | Class    | Accuracy | Precision | Recall | F1-score |
|----------|----------|----------|-----------|--------|----------|
| CNN      | COVID    | 97.83    | 100       | 97.83  | 98.90    |
|          | Normal   | 100      | 97.87     | 100    | 98.92    |
| RF       | COVID    | 95.65    | 100       | 95.65  | 97.78    |
|          | Normal   | 100      | 95.83     | 100    | 97.87    |
| DT       | COVID    | 96.09    | 100       | 96.09  | 98.00    |
|          | Normal   | 100      | 96.23     | 100    | 98.08    |
| SVM      | COVID    | 96.09    | 95.22     | 96.09  | 95.65    |
|          | Normal   | 95.22    | 96.05     | 95.22  | 95.63    |
| AB       | COVID    | 96.52    | 100       | 96.52  | 98.23    |
|          | Normal   | 100      | 96.64     | 100    | 98.29    |
| Ensemble | COVID    | 98.70    | 100       | 98.70  | 99.34    |
| Boosted  | Normal   | 100      | 98.71     | 100    | 99.35    |
Melanoma photos and their different phases can be classified using the suggested technique in the future. Furthermore, it can be used to treat a wide range of malignancies, including brain tumours.

Compliance with ethical standards

Funding: This study was not funded by any Funding Agency.

Conflict of Interest: Both authors do not have any conflict of interest.

Ethical approval: As this study was fully involved in Mathematical and Computational studies, there are no possibilities of use of human or animals for experimental purposes.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Anthimopoulos, M., Christodoulidou, S., Ebner, L., Christe, A., & Mougakakos, S. (2016, May). Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. IEEE Transactions on Medical Imaging, 35(5), 1207–1216.

Bhagavathula, A. S., Althaleri, W. A., Rahman, J., Mahabadi, M. A., & Bandari, D. K. (2020). Knowledge and perceptions of COVID-19 among health care workers: Cross sectional study. JMRM Public Health and Surveillance, 6(2), E19160. 10.2196/19160.

Campo, M. I., Pascau, J., & Estépar, R. S. J. S. (2018). Emphysis quantification on simulated X-rays through deep learning techniques. In Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI) (pp. 273–276).

Chandra, T. B., & Verma, K. (2020). Pneumonia Detection on Chest X-Ray Using Machine Learning Paradigm. In B. B. Chaudhuri, M. Nakagawa, P. Ibarra, & S. Kumar (Eds.), Proceedings of the third international conference on computer vision & image processing (pp. 21–33).

Chandra, T. B., Verma, K., Jain, D., & Némat, S. S. (2020). Localization of the suspected abnormal region in chest radiographs. In 2020 First International conference on power, control and computing technologies (ICPCCT) (pp. 204–209).

Chohan, V., Singh, S. K., Khamparia, A., Gupta, D., Tiwari, P., Moreira, C., et al. (2020, January). A novel transfer learning based approach for pneumonia detection in chest X-ray images. Applied science, 10(2), 559.

Chirigati, D. G., Giusti, A., Gambardella, L. M., & Schmidthuber, J. (2013). Mitosis detection in breast cancer histology images with deep neural networks. In Lect notes computer science: 8150 LNICS (pp. 411–414). m10.1007/978-3-642-40753-551.

Cohen, J. P., Morrison, P., Dao, L., Roth, K., Duong, T. Q., & Ghassemi, M. (2020). COVID-19 chest X-ray data collection: How the predictions are the future. [Online]. Available: http://arxiv.org/abs/2006.11988

Corman, Victor M., Landt, Olffert, Kaiser, Marco, Molenkamp, Richard, Meijer, Adam, Chua, Daniel K. W., et al. (2020). Detection of 2019 novel coronavirus (2019-nCoV) by real-time RT-PCR. Eurosurveillance, 25(3). 10.2807/1560-7917.ES.2020.25.3.2000045.

Corum, J., & Zimmer, C. (2020). Coronavirus vaccine tracker - the New york times. The New York Times. Accessed August 30, 2020. https://www.nytimes.com/interactive/2020/science/coronavirus-vaccine-tracker.html

Dong, E., Du, H., & Gardner, L. (2020). An interactive web-based dashboard to track COVID-19 in real-time. The Lancet Infectious Diseases, 20, 533–534.

He, K., X., Z., Shaoqing, R., & Jian, S. (2016). IEEE 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - Las Vegas, NV, USA (2016.6.27-2016.6.30). 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) - Deep Residual Learning for Image Recognition., 770–778. doi:10.1109/CVPR.2016.90.

Hwang, Eui Jin, Park, Sungyung, Jin, Kwang-Nam, Kim, Jung Im, Choi, So Young, Lee, Jong Hyuk, et al. (2019). Development and validation of a deep learning-based automated detection algorithm for major thoracic diseases on chest radiographs. JAMA Network Open, 2(3). Article E191095 2019. 10.1001/jamanetworkopen.2019.1095.

Islam, M. Z., Islam, M. M., & Asraf, A. (2020). A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images. Informatics in Medicine Unlocked, 20, Article 100412.

Jainwal, A. K., Tiwari, P., Kumar, S., Gupta, D., Khanna, A., & Rodrigues, J. P. J. P. C. (2019, October). Identifying pneumonia in chest X-rays: A deep learning approach. Measurement, 145, 511–518.

Ke, Q., Zhang, J., Wei, W., Popah, D., Wo’ziak, M., Ko’ziimer, L., et al. (2019). A neuro-semantic approach for recognition of lung diseases from X-ray images. Expert Systems with Applications, 126, 218–232.

Kieu, P. N., Tran, H. S., Le, T. H., Le, T., & Nguyen, T. T. (2018). Applying Multi-CNNs model for detecting abnormal problems on chest x-ray images. In Proceedings of 2018 10th international conference on knowledge and systems engineering, KSE 2018: 2018 (pp. 300–305). 10.1109/KSE.2018.8579404.

Kim, M., Yun, J., Cho, Y., Shin, K., Kang, R., Bae, H. J., et al. (2019). Deep learning in medical imaging. Neuroupin, 16(4), 657–668.
Lauer, Stephen A., Grantz, Kyra H., Bi, Qifang, Jones, Forrest K., Zheng, Qula, Meredith, Hannah R., et al. (2020). The incubation period of coronavirus disease 2019 (COVID-19) from publicly reported confirmed cases: Estimation and application. Annals of Internal Medicine, 172(9), 577–582 2020. 10.7326/M20-0504.
Li, M. D., Arun, N. T., Gidwani, M., Chang, K., Deng, F., Little, P., et al. (Jul. 2020). Automated assessment and tracking of COVID-19 pulmonary disease severity on chest radiographs using convolutional siamese neural networks. Radiology: Artificial Intelligence, 2(4), Article e200079 Art. no.
Manoj, M., Srivastava, G., Somayaji, S. R. K., Gadekallu, T. R., Maddikunta, P. K. R., & Bhattacharya, S. (2020). An incentive based approach for COVID-19 planning using blockchain technology. In 2020 IEEE Globecom Workshops (GC Wkshps): 2020 (pp. 1–6). 10.1109/GCWkshps50303.2020.9367469.
Mohsen, H., El-Dhalhan, E.-A., El-Horbati, E.-S. M., & Salem, A.-B. M. (2018). Classification using deep learning neural networks for brain tumours. Future Computer Informatics Medicine, 3(1), 68–71. 10.1016/j.fcim.2017.12.001.
Monsli, M. M. A., Poon, J., & Chung, V. (2020). Deep learning in generating radiology reports: A survey. Artificial Intelligence in Medicine, 106, Article 101878 Art. no.
Moura, et al. (2020). Deep convolutional approaches for the analysis of COVID-19 using chest X-ray images from portable devices. IEEE access: practical innovations, open solutions, 8, 195594–195607. 10.1109/ACCESS.2020.3033762.
Ozturk, T., Talo, M., Yildirim, E. A., Baloglu, U. B., Yildirim, O., & Acharya, U. R. (2020, June). Automated detection of COVID-19 cases using deep neural networks with X-ray images. Computers in biology and medicine, 121, Article 103792 Art. no.
Parsa, P., Golkov, V., Pfeiffer, F., Creners, D., & Pfeiffer, D. (2019, December). Efficient deep network architectures for fast chest X-ray tuberculosis screening and visualization. Science Reports, 9(1), Article 6268 Art. no.
Quang, D., Xie, X., & Dan, Q. (2016). A hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Research, 44(11) 2016. 10.1093/nar/gkw226.
Rajaraman, S., Siegelman, J., Alderson, P. O., Folio, L. S., Folio, L. R. & & Antani, S. K. (2020). Iteratively Pruned Deep Learning Ensembles for COVID-19 Detection in Chest X-rays. IEEE access: practical innovations, open solutions, 8, 115041–115050. 10.1109/ACCESS.2020.3003810.
Rastgarpour, M., & Shanbehzadeh, J. (2011). Application of AI techniques in medical image segmentation and novel categorization of available methods and tools. In IMECS2011: International Multi-Conference of Engineers and Computer Scientists: 1 (pp. 519–523).
Ronneberger, O., Fischer, P., & Brox, T. (2015). Convolutional networks for biomedical image segmentation. In International Conference on MedicalImage Computing and Computer-Assisted Intervention (pp. 234–241). Springer.
Saha, P., et al. (2018). NIH chest X-rays. Kaggle, 2018. [Online]. Available: Https://www.kaggle.com/nih-chest-xrays/data.