Design and Development of Efficient Medical Image Analysis Techniques

Enrolment Number(s)	-9917103079, 9917103098, 9917103134
Name of Student(s)	-Apurav Sharma, Sawan Kumar, Krishna Bharadwaj
Name of Supervisor	-Dr Mukesh Saraswat

December 2020

Submitted in partial fulfillment of the Degree of

Bachelor of Technology

In

Computer Science Engineering

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

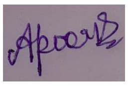
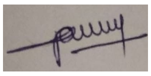
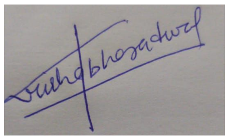
TABLE OF CONTENTS

DECLARATION

I/We hereby declare that this submission is my/our own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Place(s): Ghaziabad

Date: 29-11-2020

Signature(s): , , 

Name(s): Apurav Sharma, Sawan Kumar, Krishna Bharadwaj

Enrollment No(s): 9917103079, 9917103098, 9917103134

CERTIFICATE

This is to certify that the work titled "**Design and development of efficient medical image analysis techniques**" submitted by "**Apurav Sharma, Sawan Kumar and Krishna Bharadwaj**" in partial fulfillment for the award of degree of **Bachelor of Technology** of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of Supervisor: ……………………..

Name of Supervisor: Dr Mukesh Saraswat

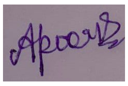
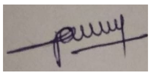
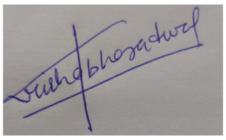
Designation: Associate Professor

Date: ……………………..

ACKNOWLEDGEMENT

I would like to place on record my deep sense of gratitude to Dr. Mukesh Saraswat, Associate Professor, Jaypee Institute of Information Technology, India for his generous guidance, help and useful suggestions.

I express my sincere gratitude to the Computer Science and Engineering Dept. of Jaypee Institute of Information Technology, India, for his/her stimulating guidance, continuous encouragement and supervision throughout the course of present work.

Signature(s):  ,  , 

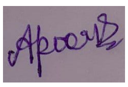
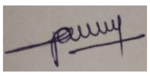
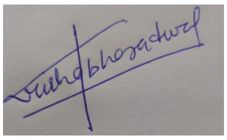
Name(s): Apurav Sharma, Sawan Kumar, Krishna Bharadwaj

Enrollment No(s): 9917103079, 9917103098, 9917103134

Date: 29-11-2020

SUMMARY

The novel coronavirus 2019 (COVID-2019), which first appeared in Wuhan city of China in December 2019, spread rapidly around the world and became a pandemic. It has caused a devastating effect on daily lives, public health and the global economy. It is critical to detect the positive cases as early as possible so as to prevent the further spread of this epidemic and to quickly treat affected patients. The need for auxiliary diagnostic tools has increased as there are no accurate automated toolkits available. Recent findings obtained using radiology imaging techniques suggest that such images contain salient information about the COVID-19 virus. Application of advanced artificial intelligence (AI) techniques coupled with radiological imaging can be helpful for the accurate detection of this disease, and can also be assistive to overcome the problem of a lack of specialized physicians in remote villages. In this study, a new model for automatic COVID-19 detection using raw chest X-ray images is presented. The proposed model is developed to provide accurate diagnostics for binary classification (COVID vs. No-Findings). Our DarkCovidNet Modified model produced a classification accuracy of 94.20% for binary classes on 625 images. We implemented 6+6+6 convolutional layers with different filtering, batch normalization and pooling. Our model can be employed to assist radiologists in validating their initial screening, and can also be employed via cloud to immediately screen patients.

Apurav Sharma, Sawan Kumar, Krishna Bharadwaj Dr Mukesh Saraswat

29-11-2020

List of Figures

List of Tables

LIST OF SYMBOLS & ACRONYMS

ResNet - Residual Network

Acc - accuracy

- Number of

Cnn - Convolutional Neural Network

Val_acc - validation accuracy

Val_loss - validation loss

DL - deep learning

1. Introduction

Recently, many radiology images have been widely used for COVD-19 detection. Hemdan et al. [1] used deep learning models to diagnose COVID-19 in X-ray images and proposed a COVIDX-Net model comprising seven CNN models. Wang and Wong [2] proposed a deep model for COVID19 detection (COVID-Net), which obtained 92.4% accuracy in classifying normal, non-COVID pneumonia, and COVID-19 classes. Ioannis et al. [3] developed the deep learning model using 224 confirmed COVID-19 images. Their model achieved 98.75% and 93.48% success rates for two and three classes, respectively. Narin et al. [4] achieved a 98% COVID-19 detection accuracy using chest X-ray images coupled with the ResNet50 model. Sethy and Behera [5] classified the features obtained from various convolutional neural network (CNN) models with support vector machine (SVM) classifier using X-ray images. Their study states that the ResNet50 model with SVM classifier provided the best performance. Finally, there are also several recent studies on COVID-19 detection that employed various deep learning models with CT images [6–9].

In this study, a deep learning model is proposed for the automatic diagnosis of COVID-19. The proposed model requires raw chest X-ray images to return the diagnosis. This model is trained with 418 chest X-ray images, which are not in a regular form and were obtained hastily. Diagnostic tests performed after 5–13 days are found to be positive in recovered patients [12]. This crucial finding shows us that recovered patients may continue to spread the virus. Therefore, more accurate methods for diagnosis are needed. One of the most important disadvantages of chest radiography analyses is an inability to detect the early stages of COVID-19, as they do not have sufficient sensitivity in GGO detection. However, well-trained deep learning models can focus on points that are not noticeable to the human eye, and may serve to reverse this perception. We obtained our datasets from[1]. In particular, our dataset consists of 1022 images for 'Covid-19' , 'No_findings' , 'Pneumonia' as diseases.

1.2 Problem Statement

The project aims to design and develop efficient medical image analysis techniques using image preprocessing. The aim of this project is to provide the techniques for classification of images for faster and cheaper diagnosis. We aim to achieve these techniques by using different deep learning models. As the deep learning models require a large amount of data for training and testing, we aim to use multiple and larger datasets in future.

1.3 Significance of the problem

Medical image processing plays a crucial role in many clinical scenarios, including in diagnosis and treatment planning, But immense quantities of data and high complexity of the algorithms often used are computationally demanding. Hence there is a need to evolve some efficient techniques which can be accelerated by high-performance computing solutions. At present, a real-time reverse-transcription polymerase chain reaction (RT-PCR) assay for COVID-19 has been developed and used in clinics. Although RT-PCR remains the reference standard for making a definitive diagnosis of COVID-19 infection, the high false-negative rate and the unavailability of the RT-PCR assay in the early stage of the outbreak restricted prompt diagnosis of infected patients.

1.4 Empirical study

Deep learning techniques have been successfully applied in many problems such as arrhythmia detection [17-19], skin cancer classification [20,21], breast cancer detection [22.23], brain disease classification [24], pneumonia detection from chest X-ray images [25], fundus image segmentation [26], and lung segmentation [27,28].

1.5 Brief Description of the Solution Approach

Instead of initiating a deep model development from scratch, a more rational approach is to construct a model using already proven models. Therefore, while designing the deep model used in this study, the DarkCovidNet model [10] is chosen as the starting point. DarkCovidNet is the classifier model. This system has the state-of-the-art architecture designed for diagnosis of COVID-19. The proposed model has 21 convolution layers. For comparison purposes we have used three pretrained models ie.InceptionV3, ResNet50 & VGG16. These models have been trained on an imagenet dataset containing more than 14million images with 20 thousands categories.

1.6 Comparison of existing approaches to the problem framed

Diagnostic tests performed even after months of covid outbreak are found to be positive in recovered patients. This crucial finding shows us that recovered patients may continue to spread the virus. Therefore, more accurate methods for diagnosis are needed. The pre-trained models are time consuming as they contain a large number of hidden layers and hence preparing a model from scratch would be a better option.

2. Literature Survey

2.1 Summary of papers studied

The work in the field of medical image analysis includes gathering data in form of images and then applying efficient models to figure out the severity of disease and it is a proven way in which patients can recover fast if detected at an earlier stage. Most of the papers signify the use of raw chest X-ray and chest CT scan images as a dataset on which the different models are implemented. Models such as COVIDX-Net, ResNet50, inception-ResNetV2 are used which are pre-trained models due to which some weights are initially assigned to the first layer and hence it sometimes deviates from the accurate result. But, instead if the model is trained from scratch then, the probability of producing more accurate results increases. However, training from scratch also increases the computational time and sometimes proves to be less efficient. Models like DRE-Net and M-Inception are also proposed but comparatively these models result in less accuracy.

Integrated summary of the literature studied is as followed:

- The aim of this article[1] is to introduce a new deep learning framework but due to the lack of public COVID-19 datasets, the study is validated on 50 Chest X-ray images with 25 confirmed positive COVID-19 cases. The COVIDX-Net includes seven different architectures of deep convolutional neural network models, such as modified Visual Geometry Group Network (VGG19) and the second version of Google MobileNet.

- In this article[2] COVID-Net is used which is a deep convolutional neural network design tailored for the detection of COVID-19 cases from chest X-ray (CXR) images that is open source and available to the general public. COVID-Net is one of the first open source network designs for COVID-19 detection from CXR images at the time of initial release.repositories. Moreover, it is investigated how COVID-Net makes predictions using an explainability method in an attempt to gain deeper insights into critical factors associated with COVID cases.

- In this study[3], a dataset of X-Ray images from patients with common pneumonia, Covid-19, and normal incidents was utilized for the automatic detection of the Coronavirus.The aim of the study is to evaluate the performance of state-of-the-art Convolutional Neural Network architectures proposed over recent years for medical image classification.It is demonstrated that the transfer learning strategy with CNNs can have significant effects on the automatic detection and automatic extraction of essential features from X-ray images, related to the diagnosis of the Covid-19.

- In this study[4], three different convolutional neural network based models (ResNet50, InceptionV3 and Inception-ResNetV2) have been proposed for the detection of coronavirus pneumonia infected patients using chest X-ray radiographs. ROC analyses and confusion matrices by these three models are given and analyzed using 5-fold cross validation. It is seen that the pre-trained ResNet50 model provides the highest classification performance with 98% accuracy among other two proposed models.

- In this paper[5], the deep learning based methodology is suggested for detection of coronavirus infected patients using X-ray images. The support vector machine classifies the corona affected X-ray images from others using the deep feature. The methodology is beneficial for the medical practitioner for diagnosis of coronavirus infected patients. The suggested classification model,i.e. resnet50 plus SVM achieved accuracy, FPR, F1 score, MCC and Kappa are 95.38%,95.52%, 91.41% and 90.76% respectively for detecting COVID-19.

- Chest CT scans of 88 patients diagnosed with the COVID-19[6] from hospitals of two provinces in China, 101 patients infected with bacteria pneumonia, and 86 healthy persons for comparison and modeling. Based on the collected dataset, a deep learning-based CT diagnosis system (DeepPneumonia) was developed to identify patients with COVID-19.

- Based on COVID-19 radiographic changes in CT images, it[7] hypothesized that Artificial Intelligence’s deep learning methods might be able to extract COVID-19’s specific graphical features and provide a clinical diagnosis ahead of the pathogenic test, thus saving critical time for disease control.

- A weakly-supervised deep learning-based software system[8] was developed using 3D CT volumes to detect COVID-19. For each patient, the lung region was segmented using a pre-trained UNet; then the segmented 3D lung region was fed into a 3D deep neural network to predict the probability of COVID-19 infectious. When using a probability threshold of 0.5 to classify COVID-positive and COVID-negative, the algorithm obtained an accuracy of 0.901.

- This study[9] aimed to establish an early screening model to distin-guish COVID-19 pneumonia from IAVP and healthy cases through pulmonary CT images using deep learning techniques. Models with a location-attention mechanism can classify COVID-19, IAVP, and healthy cases with an overall accuracy rate of 86.7%, and would be a promising sup-plementary diagnostic method for frontline clinical doctors.

- In this study[10], a deep learning based model is proposed to detect and classify COVID-19 cases from X-ray images. This model is fully automated with an end-to-end structure without the need for manual feature extraction.radiologists. Also, such models can be used to diagnose other chest-related diseases including tuberculosis and pneumonia. A limitation of the study is the use of a limited number of COVID-19 X-ray images.

- This study[11] introduces machine learning algorithms as applied to medical image analysis, focusing on convolutional neural networks, and emphasizing clinical aspects of the field.This study covers key research areas and applications of medical image classification, localization, detection,segmentation and registration. The paper concludes by discussing research obstacles, emerging trends and possible future directions.
-
- In this article[12], a review of the recent literature on applying deep learning technologies to advance the health care domain is given. Based on the analyzed work, it is suggested that deep learning approaches could be the vehicle for translating big biomedical data into improved human health.

- In this paper[13], a deep domain adaptation method for the diagnosis of COVID-19 (namely COVID-DA) is proposed, which aims to transfer the domain knowledge from the well-labeled source domain (i.e., typical pneumonia) to the partially-labeled target domain (i.e., COVID-19).

- In this study[14], a model is presented for medical image modality classification that is composed of three CNNs with different depths, which are combined by weighted averaging of the prediction probabilities. The proposed transfer learning method can benefit from generic features captured by CNNs pre-trained on ImageNet, and domain-specific features captured by the top layers of extremely deep CNNs and another “shallower” CNN, which are trained from scratch on medical images.

- In this study[15], the coronavirus image set has different types of images, which were acquired with different CT tools. Therefore, five feature extraction methods were utilized to find the feature set that separates the infected patches with a high accuracy. The dataset in this study was formed manually and achieved 99.68% classification accuracy. The proposed method should be tested on another coronavirus CT image dataset.

3. Requirement Analysis and Solution Approach

3.1 Overall description of the project

The purpose of this project work is to provide a proper methodology and efficient algorithms for better diagnosis. In this world of evolving machine learning and deep learning techniques it is becoming increasingly important for us to understand and implement in the correct fashion the proper software for the covid identification purpose. A need for the increase in a low cost and low overhead system is very much in demand for the reduction of the net customer costs and survival in the market of intense competitions. So we aim at developing algorithms which will prove to be more and more fruitful for this purpose and will be of immense applicability in the medical field. The employment of newer concepts and algorithms will prove to be a much better headway in this field.

3.2 Requirement Analysis

The users who are developing this whole application or are trying to make an evaluation of the proper working of the newest ideas and the algorithms implemented need to be proficient in the following areas . Deep learning including all concepts of Convolutional Neural Networks. Data augmentation techniques in different frameworks such as keras, tensor-flow. Basic Python programming with important data structures and libraries.

The whole of this project work will be developed using the google colab Notebook which uses the Python language for the purpose of the development of the network and the keras with tensor flow in the back-end for the module definitions. So the specific requirements for this project work is only Google colab cloud based platform. As an end product the algorithm developed by us will be implemented in mobile or other embedded devices. In that scenario the whole internal working will be abstracted from the actual users of the product and it is of no importance for them to understand the specific requirements of the programs executing inside their devices.

The algorithms developed will be implemented on embedded devices as an end product. So it is not our concern about the performance of the embedded devices which is rather a concern of the company who will be implementing this technology in their products. For the purpose of development the only performance requirements are related to the specific software requirements mentioned above and a system which is capable of smoothly running these above said software is good enough for the project members to work on the proposed algorithms.

3.5 Solution Approach

Instead of initiating a deep model development from scratch, a more rational approach is to construct a model using already proven models. Therefore, while designing the deep model used in this study, the DarkCovidNet model [10] is chosen as the starting point. DarkCovidNet is the classifier model [10]. This system has the state-of-the-art architecture designed for diagnosis of COVID-19. The proposed model has 21 convolution layers. In Fig. 1, each DN(DarkNet) layer has one convolutional layer with relu activation function followed by BatchNorm and max pooling, while each 3 Conv layer has the same setup three times in successive form. The batch normalization operation is used to standardize the inputs, and this operation has other benefits, such as reducing training time and increasing stability of the model. Maxpool downsizes an input by taking the maximum of a region determined by its filter. When working with two classes, the proposed model performs the COVID-19 detection task. If three different classes of images are used in the input, the same model can perform the classification task to determine the labels of the input chest X-ray images as COVID-19, Pneumonia, or No-Findings. Finally, the layer details and layer parameters of the model are given in Table 1. The developed deep learning model consists of 920,113 parameters. We have used the Adam optimizer for weight updates, cross entropy loss function and selected learning rate as 1e-3.

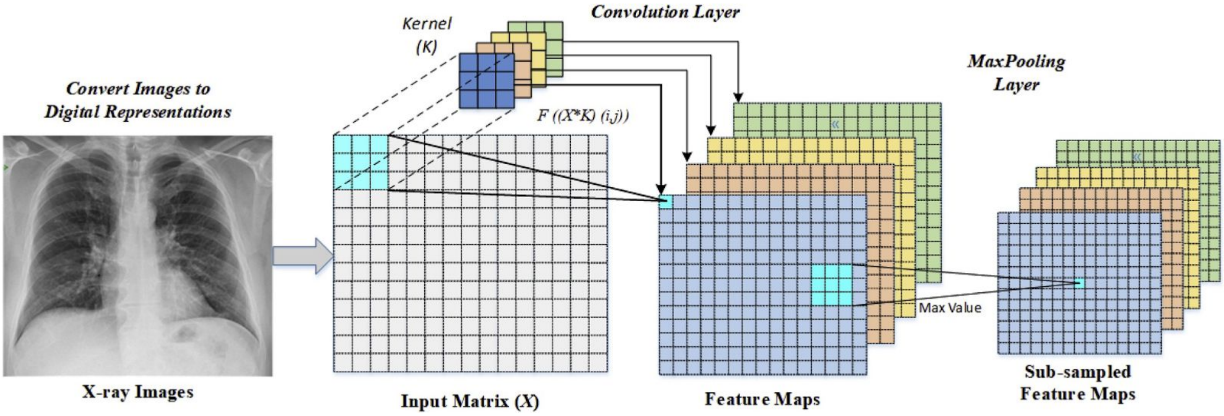


Fig 3.1 DN(DarkNet) layer

The proposed model has 21 convolution layers with figure shown below :

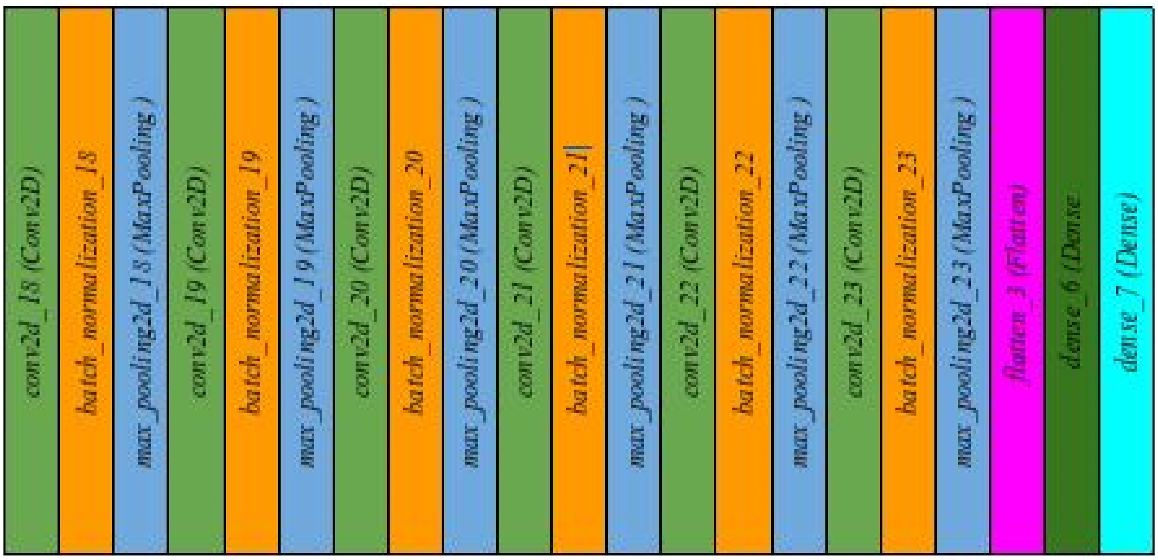


Fig 3.2 Modified DarkCovidNet

Table 1

The layers and layer parameters of the proposed model (for the binary classification task).

Layer No	Layer (type)	Output Shape	Parameters
1	conv2d_18 (Conv2D)	(256, 256, 8)	224
2	batch_normalization_18	(256, 256, 8)	32
3	max_pooling2d_18 (MaxPooling	(128, 128, 8)	0
4	conv2d_19 (Conv2D)	(128, 128, 16)	1168
5	batch_normalization_19	(128, 128, 16)	64
6	max_pooling2d_19 (MaxPooling	(64, 64, 16)	0
7	conv2d_20 (Conv2D)	(64, 64, 32)	4640
8	batch_normalization_20	(64, 64, 32)	128
9	max_pooling2d_20 (MaxPooling	(32, 32, 32)	0
10	conv2d_21 (Conv2D)	(32, 32, 64)	18496
11	batch_normalization_21	(32, 32, 64)	256
12	max_pooling2d_21 (MaxPooling	(16, 16, 64)	0
13	conv2d_22 (Conv2D)	(16, 16, 128)	73856
14	batch_normalization_22	(16, 16, 128)	512
15	max_pooling2d_22 (MaxPooling	(8, 8, 128)	0
16	conv2d_23 (Conv2D)	(8, 8, 256)	295168

17	batch_normalization_23	(8, 8, 256)	1024
18	max_pooling2d_23 (MaxPooling	(4, 4, 256)	0
19	flatten_3 (Flatten)	(4096)	0
20	dense_6 (Dense)	(128)	524416
21	dense_7 (Dense)	(1)	129

For comparison purposes we have used three pretrained models. These models have been trained on an imagenet dataset containing more than 14million images with 20 thousands categories. The models are as follows:

1. Inception v3 is a type of cnn with a depth of 159 and size of 92MB. It is a widely-used image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model is the culmination of many ideas developed by multiple researchers over the years. It is based on the original paper: "Rethinking the Inception Architecture for Computer Vision" by Szegedy, et. al[16].

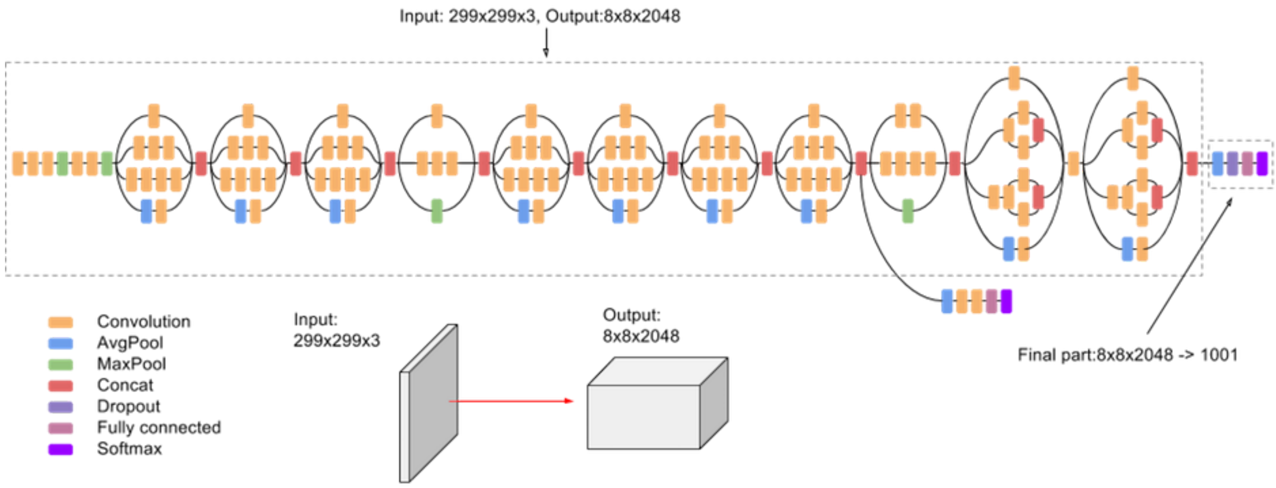


Fig 3.3 Inception v3

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concats, dropouts, and fully connected layers. Batchnorm is used extensively throughout the model and applied to activation inputs. Loss is computed via Softmax.

2. ResNet50 is a variant of the ResNet model containing 48 Convolution layers along with 1 MaxPool and 1 Average Pool layer. It has a Depth of 50 layers and a size of 98MB. It is an improved version of cnn and is also trained on imagenet dataset.

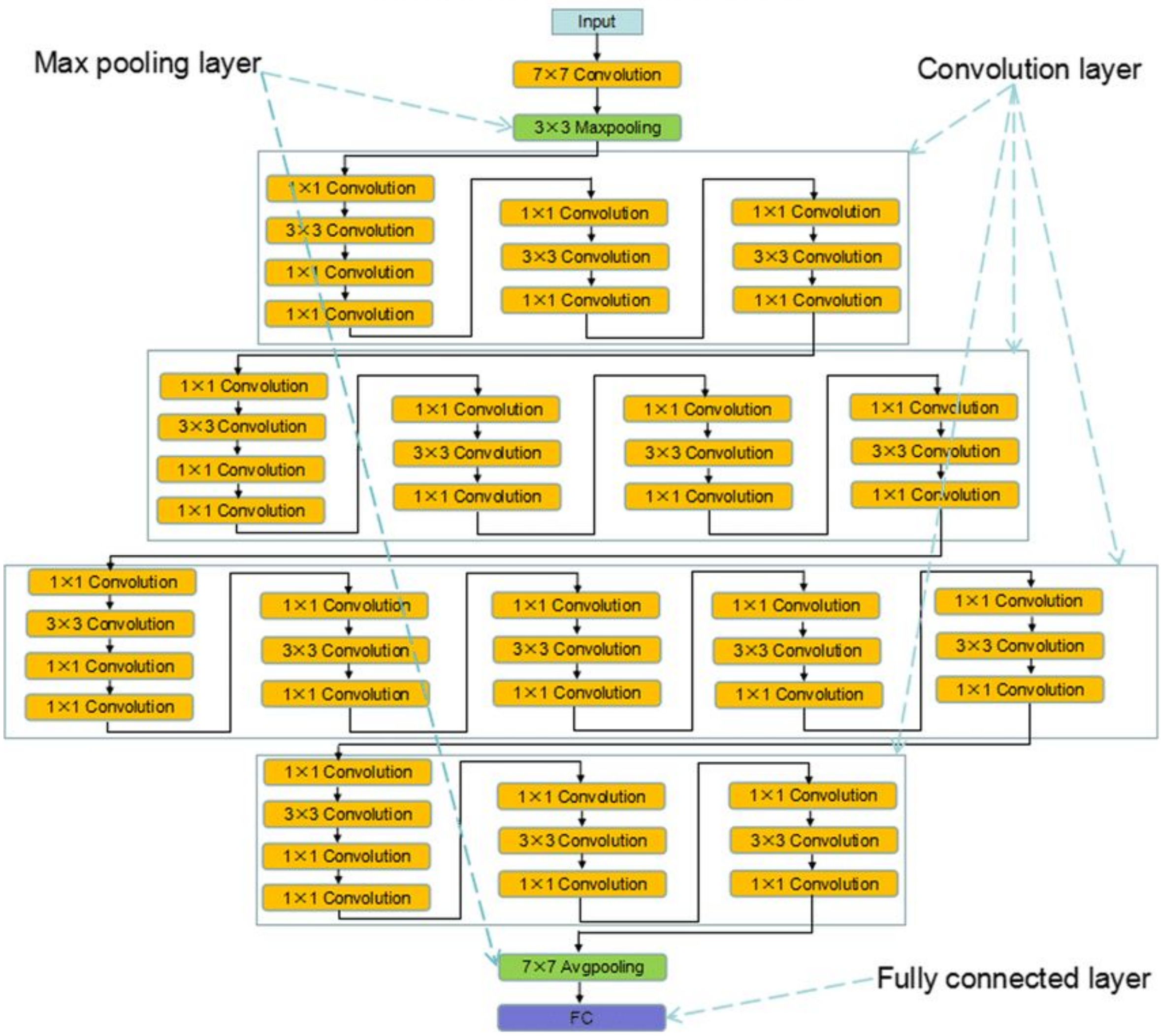


Fig 3.4 ResNet50

3. VGG16 introduced by Simonyan and Zisserman in their 2014 paper, Very Deep Convolutional Networks for Large Scale Image Recognition. It is characterized by its simplicity, using only 3×3 convolutional layers stacked on top of each other in increasing depth. It has a size of 528 MB and Depth of 23.

Fig 3.5 VGG16

4. Modeling and Implementation Details

4.1 Design Diagrams

4.1.1Use Case diagram

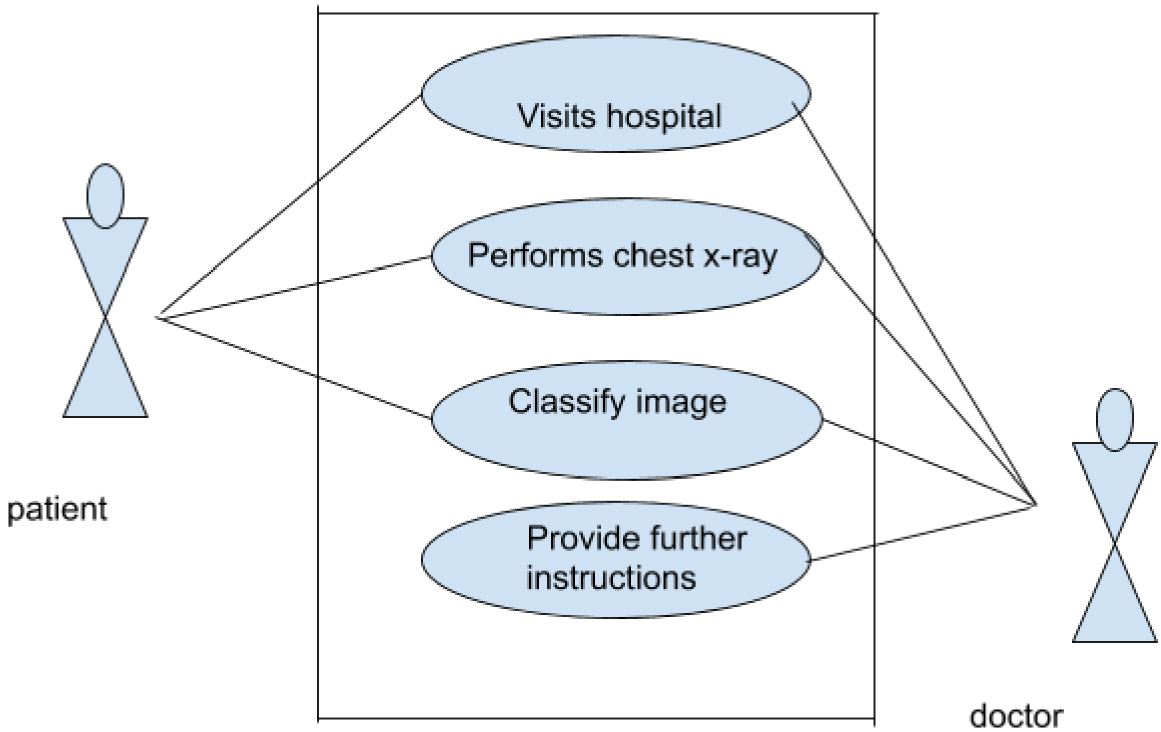


Fig 4.1 Use Case diagram

A use case diagram is a dynamic or behavior diagram in UML. The "actors" are people or entities operating under defined roles within the system. In fig 4.1 the actors are patient and doctor and the four use cases are shown in the box as a set of actions, services, and functions that the system needs to perform.

4.1.2 Class diagram / Control Flow Diagram

In fig 4.2, a class diagram shown in the Unified Modeling Language as a type of static structure diagram that describes the structure of our model by showing it's classes, their attributes, operations, and the relationships among objects. Ths diagram changes as we add or remove objects/ layers from our model and thus is drawn in a very simple and generalistic way to avoid understood changes. This diagram can also explain the relationship between different classes and objects. Different possible functions are included in the classes as methods which may or may not be used and thus lie at the discretionary power of the model such as backwardPropogate().

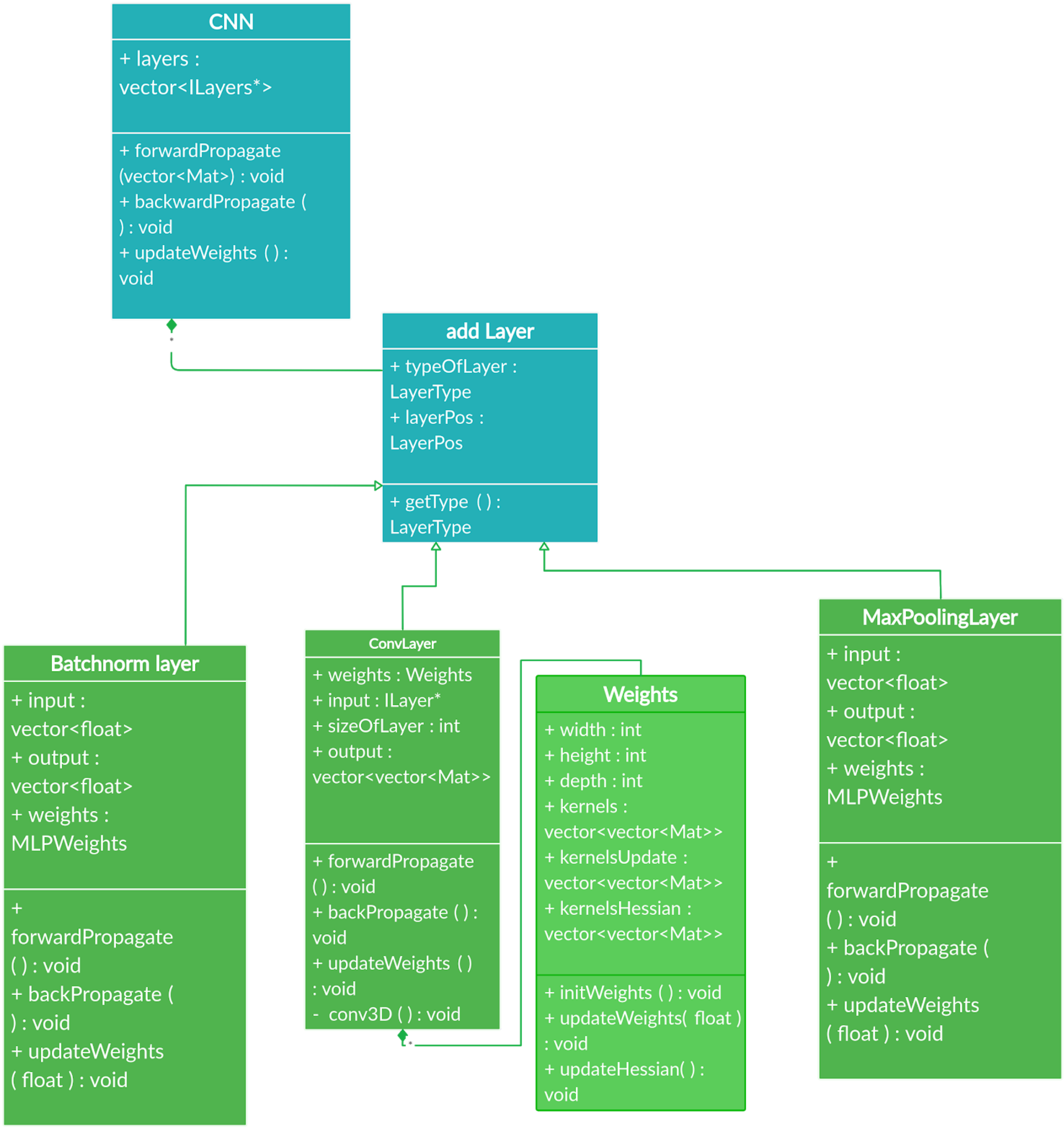


Fig 4.2 Class diagram

4.1.3 Sequence Diagram/Activity diagrams

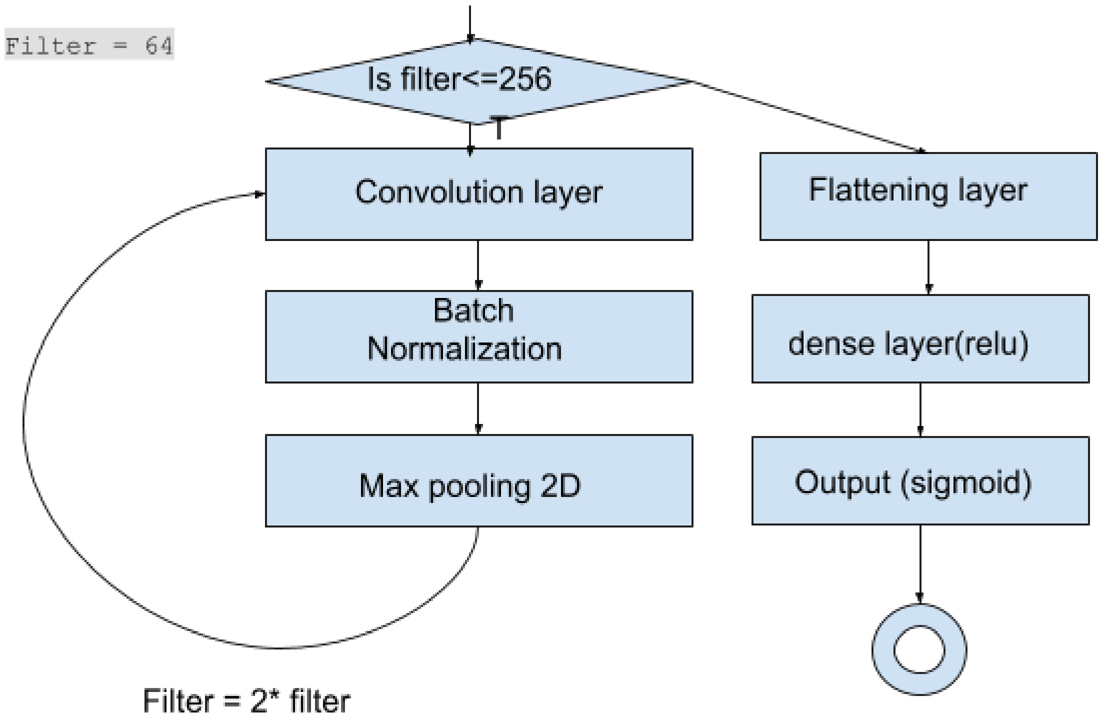


Fig 4.3 Activity diagram

The activity diagram of this model starts with a convolution layer by checking the no of filters in use. Therefore, a variable named filter depicting no of filters used for convolution layer is defined initially and assigned a value of 64. Depending upon the size of the filter, different layers of CNN are used. If filter size is less than 256 then a loop is executed which consists of a convolutional layer which further uses batch normalization and max pooling to produce output. On the other hand, for large filters a flattening layer is activated which uses relu activation function and produces output using sigmoid function. The activity diagram ends here with a stop.

4.2 Implementation details and issues

Initialization of the CNN model is the first thing to do after importing important packages. There are basically two ways to initialize a network by using Graph or Layers. We use the latter one and create an object of sequential class.

Step 1 - Convolution layer to detect features (relationships of a pixel with its neighbors) in images here we create 8 feature detectors of 3*3 dimensions, performed batch normalization and Pooling to obtain pooled/smaller feature maps.
Step 2 -Repeat step 1 for double feature detectors ie. 16, 32, 64, 128, 256
Step 3 - Flattening converts pooled feature maps to a vector/array by storing cells of these maps
Step 4 - Compiling the CNN
Step 5 – Fitting the CNN to the images

In a similar way we can implement pre-trained models using following steps:
Step 1 - create the base pre-trained model
Step 2 - add a global spatial average pooling layer
Step 3 - add a fully-connected layer and a logistic layer
Step 4 - Compiling the model
Step 5 – train the model on the new data for a few epochs

4.3 Risk Analysis and Mitigation

Google Colab is free to use and does not provide you full access to resources with the base plan . The amount of usable GPU and RAM on Colab is limited on Colab. We can try out how many scripts we can run at the same time and could start using other accounts after that. Note that inactive sessions in Colab will be closed.
I personally would try to find a way involving less computational power by keeping the models using the same dataset in a single notebook. Google Colab has got a limited amount of hardware available, and using it too much might result in other users not being able to use a GPU. Also, abusing its capacities could result in a ban for you as suggested by Jacob Schodl on stackoverflow.com.

5. Testing

5.1 Testing Plan

We have planned to use all four models on our datasets of 522 and 625 images. We will also use two different dense layers containing 128 and 1024 nodes respectively.

No of epochs = 5

When we mention validation_split as a fit parameter while fitting a DL model, it splits data into two parts for every epoch i.e. training data and validation data. It trains the model on training data and validates the model on validation data by checking its loss and accuracy.

Usually with every epoch increasing, loss goes lower and accuracy goes higher. But with val_loss and val_acc, many cases can be possible:

val_loss starts increasing, val_acc starts decreasing(means model is cramming values not learning)

val_loss starts increasing, val_acc also increases.(could be case of overfitting or diverse probability values in cases softmax is used in output layer)

val_loss starts decreasing, val_acc starts increasing(Correct, means model build is learning and working fine)

5.2 Limitations of the solution

The data-set is used as a visual aid only with 625 images and can never bring in front of our eyes the actual scenario where millions of nodes exist in a single Network. So, we must have this thing in mind that there exists a close approximation between the actual experiments developed and the practical measurements.

6. Experimental Results

6.1 Findings

We performed our results on two datasets of 522 images(using a dense layer of 128 nodes) and 625 images(including .jpg images with a dense layer of 128 and 1024 nodes, one at a time). The performance parameters used are averages of training loss, training accuracy, validation loss and validation accuracy calculated for five epochs.

A loss function is used to optimize a machine learning algorithm. The loss is calculated on training and validation and its interpretation is based on how well the model is doing in these two sets. It is the sum of errors made for each example in training or validation sets. Loss value implies how poorly or well a model behaves after each iteration of optimization. An accuracy metric is used to measure the algorithm's performance in an interpretable way.

The accuracy of a model is usually determined after the model parameters and is calculated in the form of a percentage. It is the measure of how accurate your model's prediction is compared to the true data. Training a model simply means learning (determining) good values for all the weights and the bias from labeled examples. Loss is the result of a bad prediction. A loss is a number indicating how bad the model's prediction was on a single example.

If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater. The goal of training a model is to find a set of weights and biases that have low loss, on average, across all examples.

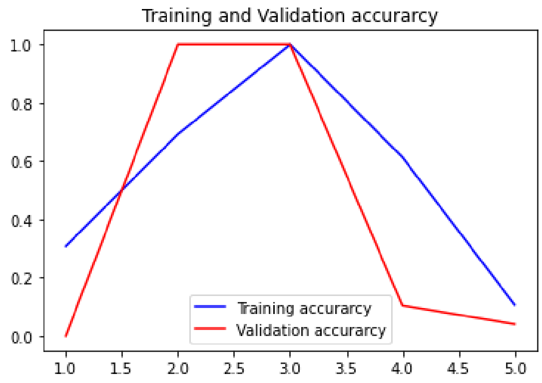
Higher loss is the worst(bad prediction) for any model. The loss is calculated on training and validation and its interpretation is how well the model is doing for these two sets. Unlike accuracy, a loss is not a percentage. It is a sum of the errors made for each example in training or validation sets[31].

Table 2

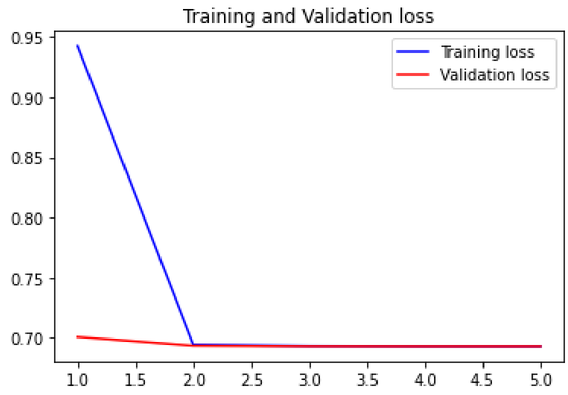
Comparison of the proposed model with other models

Model	# of Images (# of dense layer nodes)	avg training loss	avg training accuracy	avg val_loss	avg val_acc
Inception V3	522(1024)	0.6936	0.4857	0.6937	0.2476
	625 (128) 625(1024)	0.74342 0.69	0.544 0.10	0.6948 0.69	0.40 0.04
Resnet 50	522 (1024)	0.7949	0.9273	3.1703	0.0762
	625 (128) 625(1024)	0.74 0.71	0.54 0.50	0.692 0.71	0.42 0.00
VGG16	522 (1024)	0.6956	0.8727	0.7243	1.00
	625 (128) 625(1024)	7.52 0.69	0.98 0.52	7.69 0.69	1.00 0.34
Proposed	522 (1024)	0.0459	0.9870	1.3722	0.9238
	625 (128) 625(1024)	0.31 0.11	0.940 0.942	1.50 1.56	0.21 0.20

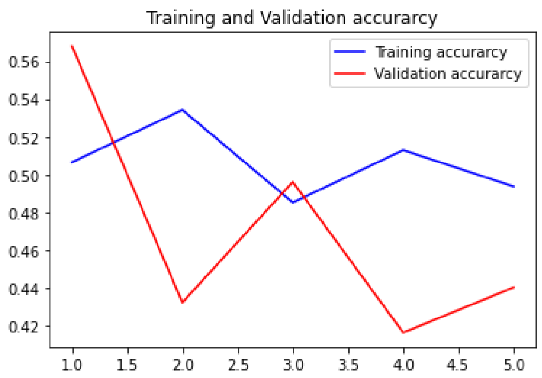
The average training loss is minimum for the updated dataset in the proposed model using 128 nodes in a dense layer and extremely high for VGG16. The average training accuracy is also abnormally high(0.98) despite high loss for the VGG16(1028 nodes in a dense layer) & 0.987 with a loss of 0.0459 in the proposed model(522 images at a dense layer of 128 nodes). Similar scenario is depicted in case of validation parameters. The VGG16 achieves a perfect validation despite abnormally high losses of 7.69 and followed by the proposed model with an average validation accuracy of 0.92 with high loss of 1.32. Greater the loss is, the more huge the errors model made on the data. Accuracy can be seen as directly proportional to the number of predictions made on the data.Thus, in case of VGG16, a great accuracy but a huge loss, means it made huge errors on a few data and the proposed model performs best among rest. Inception is also performing better than ResNet50. The same is depicted in graphs shown below.



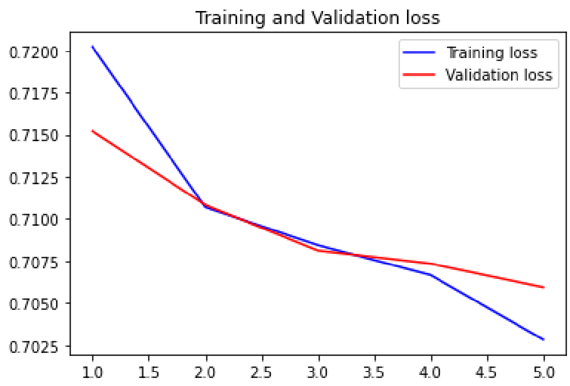
(a)



(b)



(c)

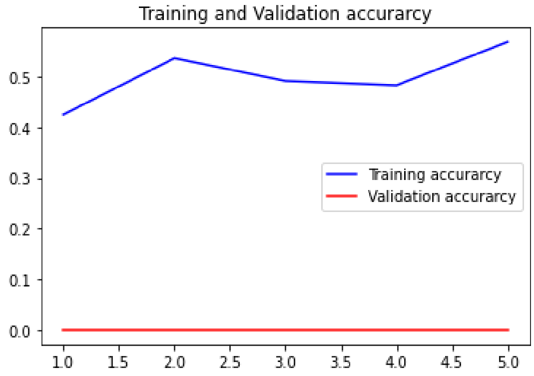


(d)

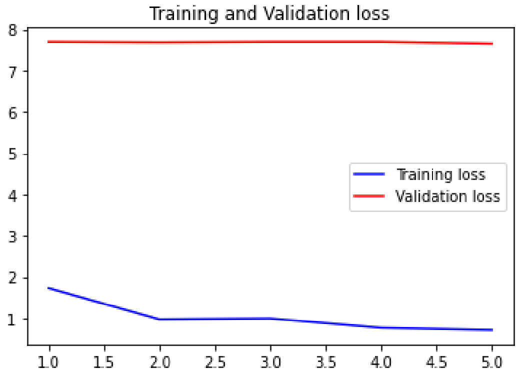
Fig 6.1 InceptionV3 Graphs

In Fig 6.1 (a) training accuracy increases with increase in no of epochs but after a certain point(after 3 epochs) training accuracy decreases. Likewise in Fig 6.1(c) training accuracy increases till 2 epoch then till 3 epoch it decreases and again increases.

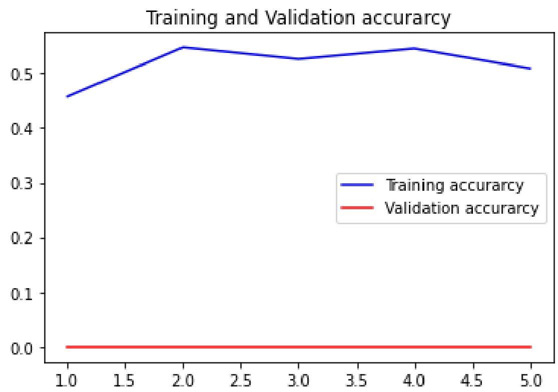
In fig 6.1(b),(d) training loss decreases with increase in number of epochs.



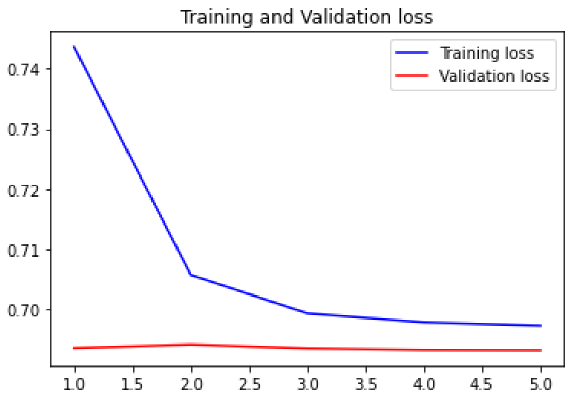
(a)



(b)



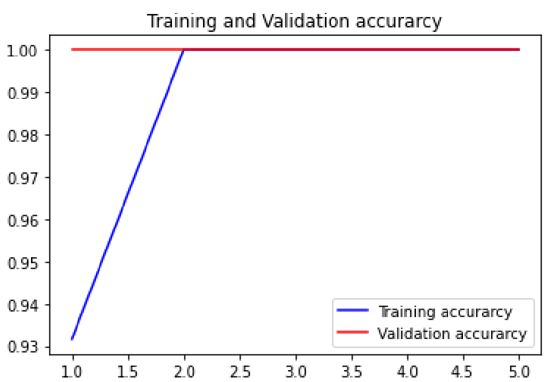
(c)



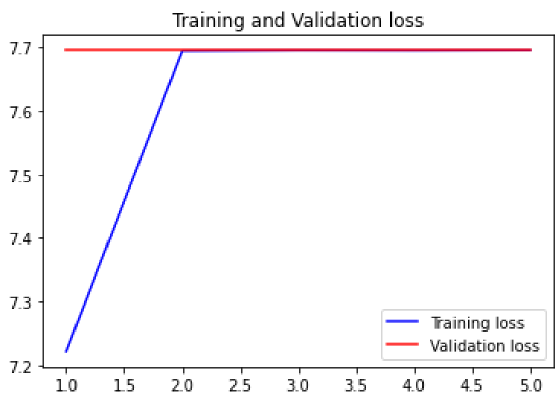
(d)

Fig 6.2 ResNet 50 Graphs

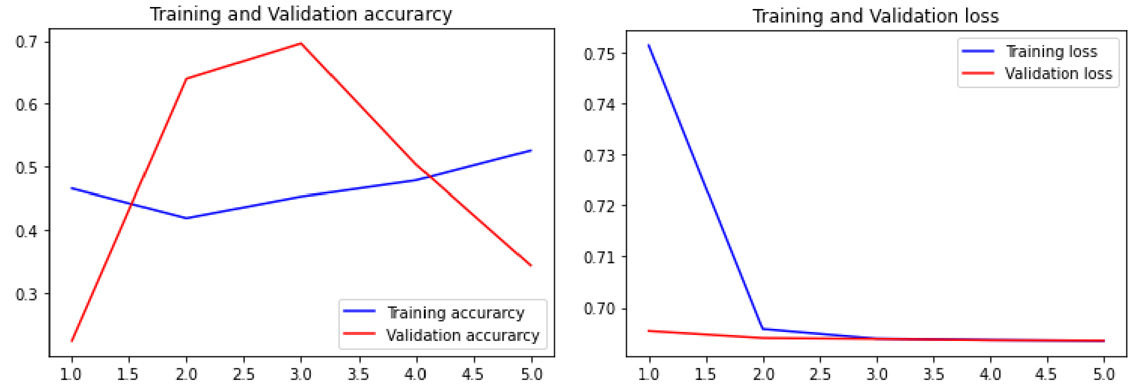
For Resnet 50, It is clearly observable that rise in training accuracy is not that gradual as compared to InceptionV3. But training loss decreases gradually like in InceptionV3.



(a)



(b)

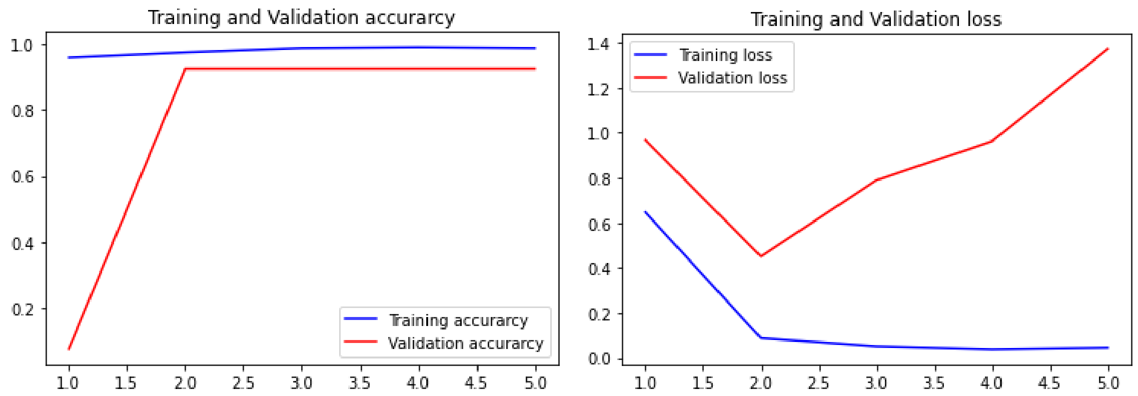


(c) (d)

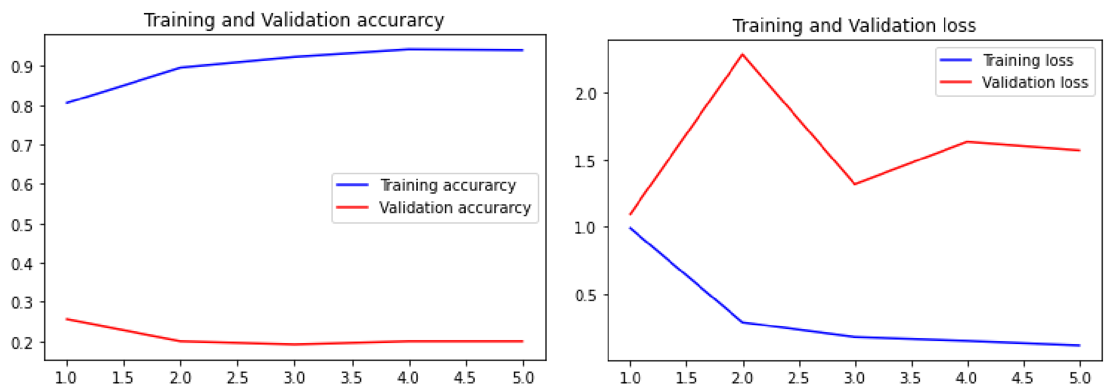
Fig 6.3 VGG16 Graphs

For VGG16, it is clear that training accuracy sharply increases in Fig 6.3(a) and training loss sharply decreases with increase in epochs in Fig 6.3(b).

In Fig 6.3(c), firstly training accuracy decreases till 2 epochs and then gradually increases. But, training loss decreases sharply till 2 epochs and then starts decreasing gradually as in Fig 6.3(d).



(a) (b)



(c) (d)

Fig 6.4 Modified DarkCovidNet Graph

For Modified DarkCovidNet, training accuracy remains constant with increase in epochs as in Fig 6.4(a) but training loss decreases sharply and then becomes constant in Fig 6.4(b).
But, In Fig 6.4(c) training accuracy increases gradually at first till 2 epochs and then becomes constant but training loss decreases gradually and then gradually becomes constant in Fig 6.4(d).

The main findings are as follows:

1. There has been a lot of focus on the type of model used or rather pre trained models have been used such as InceptionV3, Resnet50 etc. Multiple models have been cited in most of the research papers. However, most of these research models are pre trained on large amounts of data, they fail to provide good results on the dataset selected by us. We have used a modified darkcovidnet model which provides consistent and incremental performance.

2. When we use these pretrained models on our dataset they perform drastically up and down both during training and validation.

3. There is hardly any work that is performed on both types of datasets. Most of them either choose chest x-ray images or chest CT images and that too for small datasets. We have planned to do it on both types of dataset.

4. These models are time consuming as they contain a large number of hidden layers and hence preparing a model from scratch would be a better option.

5. There is hardly any research paper that performed preprocessing on the dataset images. Most of the research papers directly jump to use pre trained heavy models. We plan to perform such rituals in part two of our major project.

6.2 Conclusion

In this project, we trained a Convolutional Neural Network (CNN) Model using the Keras deep learning library. We obtained our datasets from [10]. In particular, our dataset consists of 1022 images for 'Covid-19' , 'No_findings' , 'Pneumonia' for disease data-set and we have used 625 images for binary classification. Using our Convolutional Neural Network modified Darknet, we were able to obtain 94.2% accuracy of binary classification, which is quite respectable given the limited size of our dataset. For comparison purposes we have used three pretrained models ie. InceptionV3, ResNet50 & VGG16 among which the performance of ResNet50 is poorest and InceptionV3 performs slightly better. VGG16 succumbed to abnormal loss.

6.3 Future Work

- Implement on other datasets
- Classify multiple categories
- Testing to improve accuracy
- Add more features if time permits

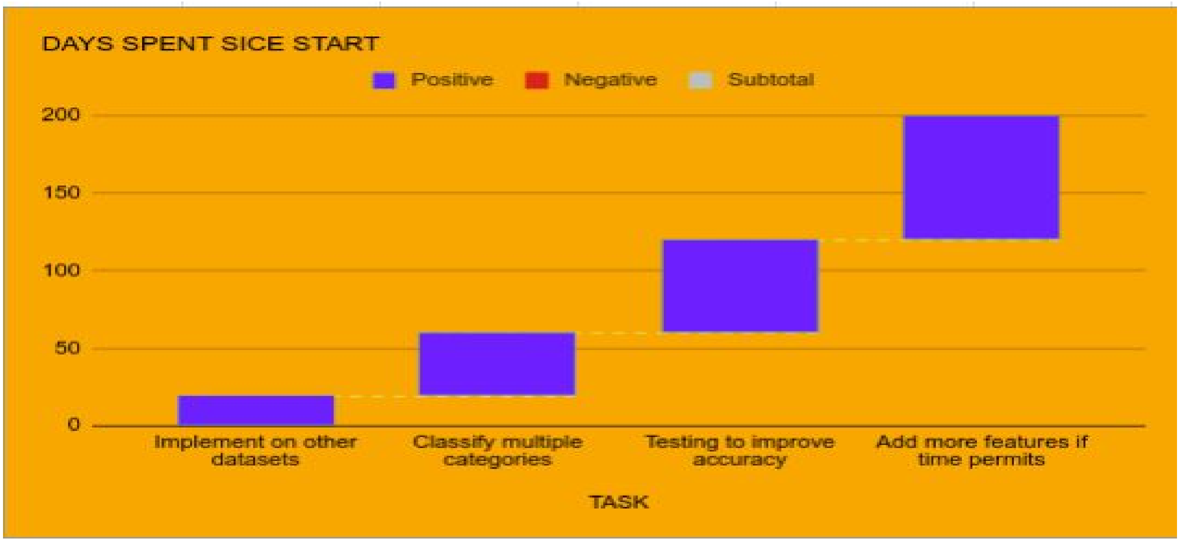


Fig:6.1 gantt chart

In future, the performance of the developed model can be assessed by expert radiologists and can be tested with a larger database and finally enable an individual to use our model and deploy it to perform diagnosis easily.

References:

[1] E.E.D. Hemdan, M.A. Shouman, M.E. Karar, COVIDX-Net: A Framework of Deep Learning Classifiers to Diagnose COVID-19 in X-Ray Images, 2020

[2] L. Wang, A. Wong, COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest Radiography Images, 2020

[3] Ioannis D. Apostolopoulos1, Tzani Bessiana, COVID-19: Automatic Detection from X-Ray Images Utilizing Transfer Learning with Convolutional Neural Networks,

[4] A. Narin, C. Kaya, Z. Pamuk, Automatic Detection of Coronavirus Disease (COVID-19) Using X-Ray Images and Deep Convolutional Neural Networks, 2020

[5] P.K. Sethy, S.K. Behera, Detection of Coronavirus Disease (COVID-19) Based on Deep Features, 2020.

[6] Y. Song, S. Zheng, L. Li, X. Zhang, X. Zhang, Z. Huang, Y. Chong, Deep learning enables accurate diagnosis of novel coronavirus (COVID-19) with CT images, medRxiv (2020).

[7] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, B. Xu, A deep learning algorithm using CT images to screen for CoronaVirus Disease (COVID-19), medRxiv (2020).

[8] C. Zheng, X. Deng, Q. Fu, Q. Zhou, J. Feng, H. Ma, X. Wang, Deep learning-based detection for COVID-19 from chest CT using weak label, medRxiv (2020), https://doi.org/10.1101/2020.03.12.20027185.

[9] X. Xu, X. Jiang, C. Ma, P. Du, X. Li, S. Lv, et al., Deep Learning System to Screen Coronavirus Disease 2019 Pneumonia, 2020

[10] Tulin Ozturk, Muhammed Talo, Eylul Azra Yildirim , Ulas Baran Baloglu, Ozal Yildirim, U. Rajendra Acharya - Automated detection of COVID-19 cases using deep neural networks with X-ray images, Computers in Biology and Medicine 121 (2020) 103792

[11] J. Ker, L. Wang, J. Rao, T. Lim, Deep learning applications in medical image analysis, Ieee Access 6 (2017) 9375–9389.

[12] Riccardo Miotto*, Fei Wang*, Shuang Wang, Xiaoqian Jiang and Joel T. Dudley ,Deep learning for healthcare: review, opportunities and challenges.

[13]Yifan Zhang, Shuaicheng Niu, Zhen Qiu, Ying Wei, Peilin Zhao, Jianhua Yao, Junzhou Huang, Qingyao Wu, and Mingkui Tan , COVID-DA: Deep Domain Adaptation from Typical Pneumonia to COVID-19.

[14] Yuhai Yu , Hongfei Lin , Jiana Meng , Xiaocong Wei , Hai Guo and Zhehuan Zhao, Deep Transfer Learning for Modality Classification of Medical Images.

[15] M. Barstugan, U. Ozkaya, S. Ozturk, Coronavirus (COVID-19) Classification Using CT Images by Machine Learning Methods, 2020

[16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Rethinking the Inception Architecture for Computer Vision, https://arxiv.org/pdf/1512.00567.pdf

[17] O.Yıldırım, P. Pławiak, R.S. Tan, U.R. Acharya, Arrhythmia detection using deep convolutional neural network with long duration ECG signals, Comput. Biol. Med. 102 (2018) 411–420.

[18] A.Y. Hannun, P. Rajpurkar, M. Haghpanahi, G.H. Tison, C. Bourn, M.P. Turakhia, A.Y. Ng, Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network, Nat. Med. 25 (1) (2019) 65.

[19] U.R. Acharya, S.L. Oh, Y. Hagiwara, J.H. Tan, M. Adam, A. Gertych, R. San Tan, A deep convolutional neural network model to classify heartbeats, Comput. Biol. Med. 89 (2017) 389–396.

[20] A. Esteva, B. Kuprel, R.A. Novoa, et al., Dermatologist-level classification of skin cancer with deep neural networks, Nature 542 (7639) (2017) 115–118, https://doi.org/10.1038/nature21056.

[21] N.C. Codella, Q.B. Nguyen, S. Pankanti, D.A. Gutman, B. Helba, A.C. Halpern, J.R. Smith, Deep learning ensembles for melanoma recognition in dermoscopy images, IBM J. Res. Dev. 61 (4/5) (2017), 5-1.

[22] Y. Celik, M. Talo, O. Yildirim, M. Karabatak, U.R. Acharya, Automated invasive ductal carcinoma detection based using deep transfer learning with whole-slide images, Pattern Recogn. Lett. 133 (2020) 232–239.

[23] A. Cruz-Roa, A. Basavanhally, et al., March). Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks, in: Medical Imaging 2014: Digital Pathology, vol. 9041, International Society for Optics and Photonics, 2014, p. 904103.

[24] M. Talo, O. Yildirim, U.B. Baloglu, G. Aydin, U.R. Acharya, Convolutional neural networks for multi-class brain disease detection using MRI images, Comput. Med. Imag. Graph. 78 (2019) 101673 images, Pattern Recogn. Lett. 133 (2020) 232–239.

[25] A. Cruz-Roa, A. Basavanhally, et al., March). Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks, in: Medical Imaging 2014: Digital Pathology, vol. 9041, International Society for Optics and Photonics, 2014, p. 904103.

[26] M. Talo, O. Yildirim, U.B. Baloglu, G. Aydin, U.R. Acharya, Convolutional neural networks for multi-class brain disease detection using MRI images, Comput. Med. Imag. Graph. 78 (2019) 101673.

[27] P. Rajpurkar, J. Irvin, et al., Chexnet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning, 2017 arXiv preprint arXiv:1711.05225.[28] J.H. Tan, H. Fujita, S. Sivaprasad, S.V. Bhandary, A.K. Rao, K.C. Chua, U. R. Acharya, Automated segmentation of exudates, haemorrhages, microaneurysms using a single convolutional neural network, Inf. Sci. 420 (2017) 66–76.

[29] G. Gaal, B. Maga, A. Lukacs, Attention U-Net Based Adversarial Architectures for Chest X-Ray Lung Segmentation, 2020 arXiv preprint arXiv:2003.10304. [30] J.C. Souza, J.O.B. Diniz, J.L. Ferreira, G.L.F. da Silva, A.C. Silva, A.C. de Paiva, An automatic method for lung segmentation and reconstruction in chest X-ray using deep neural networks, Comput. Methods Progr. Biomed. 177 (2019) 285–296.

[30] Z.Y. Zu, M.D. Jiang, P.P. Xu, W. Chen, Q.Q. Ni, G.M. Lu, L.J. Zhang, Coronavirus disease 2019 (COVID-19): a perspective from China, Radiology (2020), https://doi.org/10.1148/radiol.2020200490.

[31] How to interpret "loss" and "accuracy" for a machine learning model, https://intellipaat.com