# *Intracranial Hemorrhage Detection*

*Naveen Ojha*

# Intracranial Hemorrhage Detection.

**DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF**

Master of Technology
in
Computer Science

by


## Naveen Ojha

[ Roll No: CS-1816 ]


under the guidance of

**Dr. Sushmita Mitra.**
Professor
Machine Intelligence Unit



**Indian Statistical Institute**
**Kolkata-700108, India**


**July 2016**

*To my family and my guide*

# CERTIFICATE

This is to certify that the dissertation entitled **"Intracranial Hemorrhage Detection"** submitted by **Naveen Ojha** to Indian Statistical Institute, Kolkata, in partial fulfillment for the award of the degree of **Master of Technology in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

---

**Dr. Sushmita Mitra**
Professor,
Mac Machine Intelligence Unit,
Indian Statistical Institute,
Kolkata-700108, INDIA.

# Acknowledgments

I would like to show my highest gratitude to my advisor, *Prof. Dr. Sushmita Mitra*, Machine Intelligence Unit, Indian Statistical Institute, Kolkata, for his guidance and continuous support and encouragement. He has literally taught me how to do good research, and motivated me with great insights and innovative ideas.

I would also like to thank *Sugata Banerji*, Assistant Professor, Lake Forest College and *Subhashish Bannerjee*, Senior Research Fellow, Machine Intelligence Unit, Indian Statistical Institute, Kolkata,for his valuable suggestions and discussions.

My deepest thanks to all the teachers of Indian Statistical Institute, for their valuable suggestions and discussions which added an important dimension to my research work.

Finally, I am very much thankful to my parents and family for their everlasting supports.

Last but not the least, I would like to thank all of my friends for their help and support. I thank all those, whom I have missed out from the above list.

**Naveen Ojha**
Indian Statistical Institute
Kolkata - 700108 , India.

# Abstract

ICH is diagnosed through history, physical examination, and, most commonly, non-contrast CT examination of the brain, which discloses the anatomic bleeding location. Trauma is a common cause. In the absence of trauma, spontaneous intraparenchymal hemorrhage is a common cause associated with hypertension when found in the deep locations such as the basal ganglia, pons, or caudate nucleus. [7] Automatic triage of imaging studies using computer algorithms has the potential to detect ICH earlier, ultimately leading to improved clinical outcomes. Such a quality improvement tool could be used to automatically manage the priority for interpretation of imaging studies with presumed ICH and help optimize radiology workflow. Machine learning and computer vision are among a suite of techniques for teaching computers to learn and detect patterns. [18] We have to identify acute intracranial hemorrhage and its subtypes. In this problem a patient can have more than one sub type of ICH so this problem belongs to a Multilabel Classification Problem. We have used different models to classify the ICH images.

# Contents

# Chapter 1

# Introduction

## 1.1 Introduction

Intracranial hemorrhage (ICH) is defined as bleeding within the intracranial vault and has several subtypes depending on the anatomic location of bleeding. ICH is diagnosed through history, physical examination, and, most commonly, noncontrast CT examination of the brain, which discloses the anatomic bleeding location. Trauma is a common cause. In the absence of trauma, spontaneous intraparenchymal hemorrhage is a common cause associated with hypertension when found in the deep locations such as the basal ganglia, pons, or caudate nucleus. [7] . It's a life-threatening emergency. We should go to the emergency room right away or call emergency number if we think or know is experiencing it. [2]

## 1.2 Hemorrhage Types

Hemorrhage in the head (intracranial hemorrhage) is a relatively common condition that has many causes ranging from trauma, stroke, aneurysm, vascular malformations, high blood pressure, illicit drugs and blood clotting disorders. The neurologic consequences also vary extensively depending upon the size, type of hemorrhage and location ranging from headache to death. The role of the Radiologist is to detect the hemorrhage, characterize the hemorrhage subtype, its size and to determine if the hemorrhage might be jeopardizing critical areas of the brain that might require immediate surgery. [2] While all acute (i.e. new) hemorrhages appear dense (i.e. white) on computed tomography (CT), the primary imaging features that help Radiologists determine the subtype of hemorrhage are the location, shape and proximity to other structures ( as shown in Figure 1.1). [2] Patients may exhibit more than one type of intracranial hemorrhage, which may appear on the same image. While small hemorrhages are less morbid than large hemorrhages typically, even a small hemorrhage

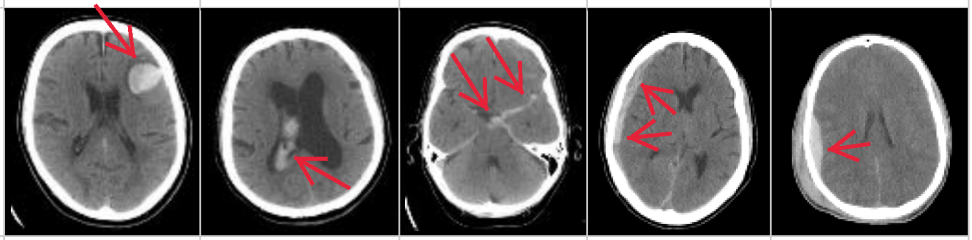
| | Intraparenchymal | Intraventricular | Subarachnoid | Subdural | Epidural |
|---|---|---|---|---|---|
| **Location** | Inside of the brain | Inside of the ventricle | Between the arachnoid and the pia mater | Between the Dura and the arachnoid | Between the dura and the skull |
| **Imaging** |  | | | | |
| **Mechanism** | High blood pressure, trauma, arteriovenous malformation, tumor, etc | Can be associated with both intraparenchymal and subarachnoid hemorrhages | Rupture of aneurysms or arteriovenous malformations or trauma | Trauma | Trauma or after surgery |
| **Source** | Arterial or venous | Arterial or venous | Predominantly arterial | Venous (bridging veins) | Arterial |
| **Shape** | Typically rounded | Conforms to ventricular shape | Tracks along the sulci and fissures | Crescent | Lentiform |
| **Presentation** | Acute (sudden onset of headache, nausea, vomiting) | Acute (sudden onset of headache, nausea, vomiting) | Acute (worst headache of life) | May be insidious (worsening headache) | Acute (skull fracture and altered mental status) |

Figure 1.1: Hemorrhage Types[2]

can lead to death because it is an indicator of another type of serious abnormality. [2]

### 1.2.1 Types of Hemorrhage

1. Intraparenchymal hemorrhage (IPH) is one form of intracerebral bleeding in which there is bleeding within brain parenchyma. The other form is intraventricular hemorrhage (IVH). Intraparenchymal hemorrhage accounts for approx. 8-13 percent of all strokes and re sults from a wide spectrum of disorders. [21]

2. Intraventricular hemorrhage (IVH), also known as intraventricular bleeding, is a bleeding into the brain's ventricular system, where the cerebrospinal fluid is produced and circulates through towards the subarachnoid space. Intraventricular extension of hemorrhage (IVH) is a particularly poor prognostic sign, with expected mortality between 50 percent and 80 percent. IVH is a significant and independent contributor to morbidity and mortality, yet therapy directed at ameliorating intraventricular clot has been limited. [9]

3. Subarachnoid hemorrhage (SAH) is bleeding into the subarachnoid space—the area between the arachnoid membrane and the pia mater surrounding the brain. Symptoms may include a severe headache of rapid onset, vomiting, decreased level of consciousness, fever, and sometimes seizures. Neck stiffness or neck pain

are also relatively common. In about a quarter of people a small bleed with resolving symptoms occurs within a month of a larger bleed. [13]

4. Subdural hemorrhage (or hematoma) is a type of bleeding that often occurs outside the brain as a result of a severe head injury. It takes place when blood vessels burst between the brain and the leather-like membrane that wraps around the brain (the dura mater). [1]

5. Epidural hematoma is when bleeding occurs between the tough outer membrane covering the brain (dura mater) and the skull. Often there is loss of consciousness following a head injury, a brief regaining of consciousness, and then loss of consciousness again. [19]

## 1.3 Problem Statement

In this problem we have to build an algorithm that can detect the Intracranial Haemorrahage and its subtypes. We are using rich image dataset provided by the Radiological Society of North America (RSNA®). This is an multilabel classification problem since one image can belong to more than one type of class. There are 6 different classes :-
i)Intraparenchymal hemorrhage
ii)Intraventricular hemorrhage
ii)Subarachnoid hemorrhage
iv)Subdural hemorrhage
v)Epidural hematoma
vi) Any of the above.

# Chapter 2

# Related Work

In these project we have worked on a new data which is availble from Kaggle Challenge RSNA Intracranial Hemorrhage Detection 2019.The aim of the project was to detect Intracranial Hemorrhage using a single slice of brain scan images. All the related work that we have seen is on 3D slice means the detecting ICH from lower to upper portion of brain scan images. In these paper [15] they have trained a fully convolutional neural network with 4,396 head CT scans performed at the University of California at San Francisco and affiliated hospitals and compared the algorithm's performance to that of 4 American Board of Radiology (ABR) certified radiologists on an independent test set of 200 randomly selected head CT scans. Their algorithm demonstrated the highest accuracy to date for this clinical application, with a receiver operating characteristic (ROC) area under the curve (AUC) of $0.991 \pm 0.006$ for identification of examinations positive for acute intracranial hemorrhage, and also exceeded the performance of 2 of 4 radiologists. They demonstrate an end-to-end network that performs joint classification and segmentation with level classification comparable to experts,in addition to robust localization of abnormalities, including some that are missed by radiologists, both of which are critically important elements for this application [15]. Since it was their data they have done annotations which simplifies their problem .We have single slice CT image of brain so our model and this papers works are not comparable.So we have focused on preprocessig and building a good model that can detect ICH.

# Chapter 3

# Data Visualisation

## 3.1 Introduction

The dataset that we have used is downloaded from kaggle website a rich image dataset provided by the Radiological Society of North America (RSNA®) in collaboration with members of the American Society of Neuroradiology and MD.ai. [2] We have been provided image in DICOM format that will contain metadata such as PatientID, StudyInstanceUID, SeriesInstanceUID, and other features. There are two csv file train.csv and test.csv in which contain the id of image and information regarding their label information .There are total five subtypes of hemorrhage and there is one more class which is any which determines whether the image belong to any of the subtypes of hemorrhage or not. We have been given training data in a such a format that each image will contain six different rows .

## 3.2 Data Description

The information regarding the label will look like [ImageId][Sub-type Name] as follows (as shown in Fig3.1)

1. Id - An image Id. Each Id corresponds to a unique image, and will contain an underscore .

2. Label - The probability of whether that sub-type of hemorrhage (or any hemorrhage in the case of any) exists in the indicated image.

There are total 4516842 training rows in train.csv.You can see the number of rows having 0 or 1 label.'0' indicates that image id will not have that kind of hemorrhage or '1' indicates that image id will have that kind of hemorrhage. So we have plot a

| ID | Label |
|---|---|
| ID_8079930a8_intraventricular | 0 |
| ID_8079930a8_subarachnoid | 0 |
| ID_8079930a8_subdural | 0 |
| ID_8079930a8_any | 0 |
| ID_4a85a3a3f_epidural | 0 |
| _4a85a3a3f_intraparenchymal | 0 |
| ID_4a85a3a3f_intraventricular | 0 |
| ID_4a85a3a3f_subarachnoid | 0 |
| ID_4a85a3a3f_subdural | 0 |
| ID_4a85a3a3f_any | 0 |

Figure 3.1: Examples of train.csv

count no of image id labelled as 0 or 1. We can see total count of images having 1 label and 0 label in Figure 3.2 .
There are two folders of images training and testing folder. There are total 755948 training images and 121232 testing images. Each image is single 2D slice having resolution 512x512 pixel.

## 3.3 Training Data distribution:

In fig 3.3 you can see the Blue bar the no of 0 label in different subtypes of intracranial hemorrhage and Orange denotes no of 1 label in different subtypes of intracranial hemorrhage . In fig 3.4 We can see count of images having that type of subtypes of intracranial hemorrhage . We can see from the fig 3.3 that there is huge data imbalance . Epidural is in the worst case.

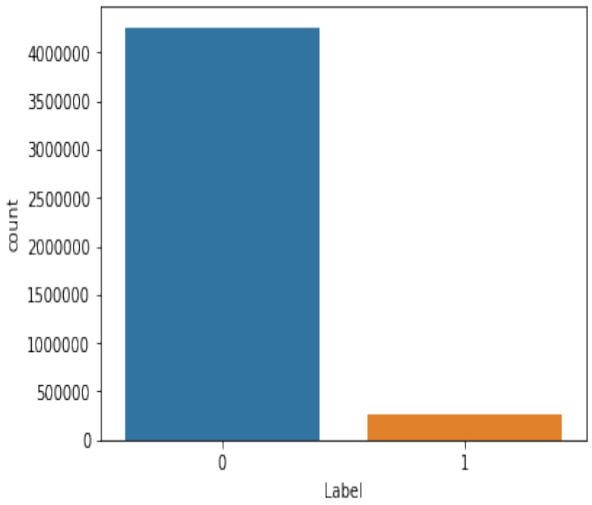



Figure 3.2: Count plot of label in train.csv

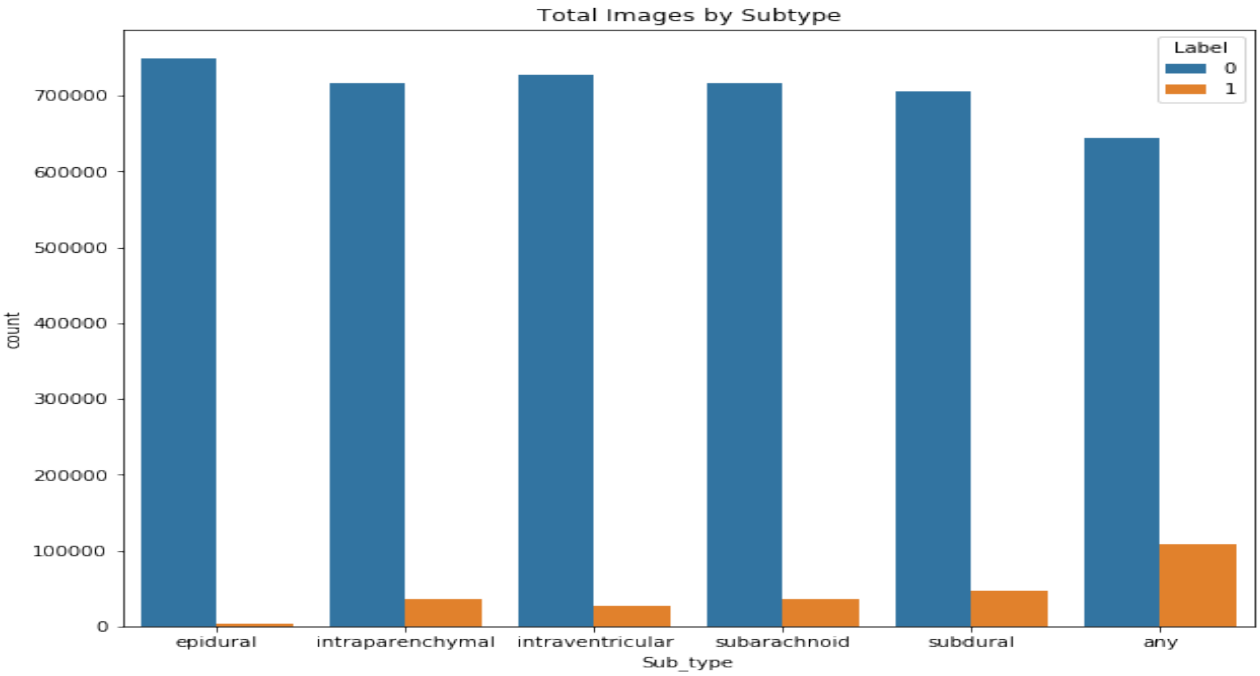



Figure 3.3: Training Data Distribution

| Sub_type | |
|---|---|
| any | 107933 |
| epidural | 3145 |
| intraparenchymal | 36118 |
| intraventricular | 26205 |
| subarachnoid | 35675 |
| subdural | 47166 |

Figure 3.4: Label count

## 3.4 Data Visualisation:

let us have look at different kind of Intracranial Hemorrhage and have some idea from those images.

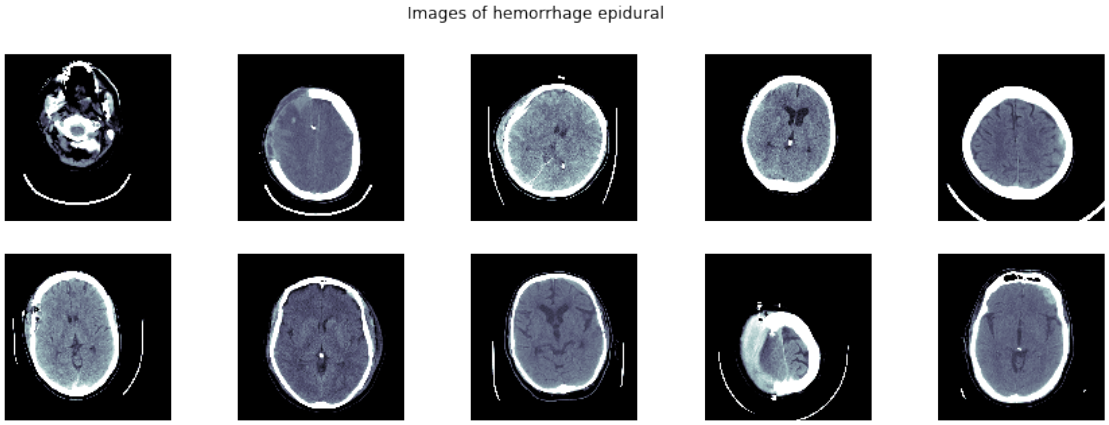



Figure 3.5: Visualis ation of Epidural Hemorrhage

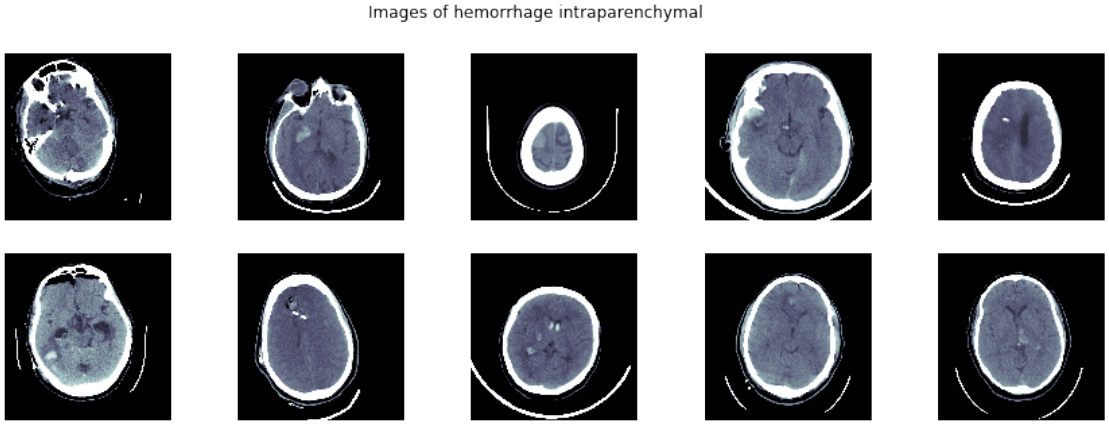



Figure 3.6

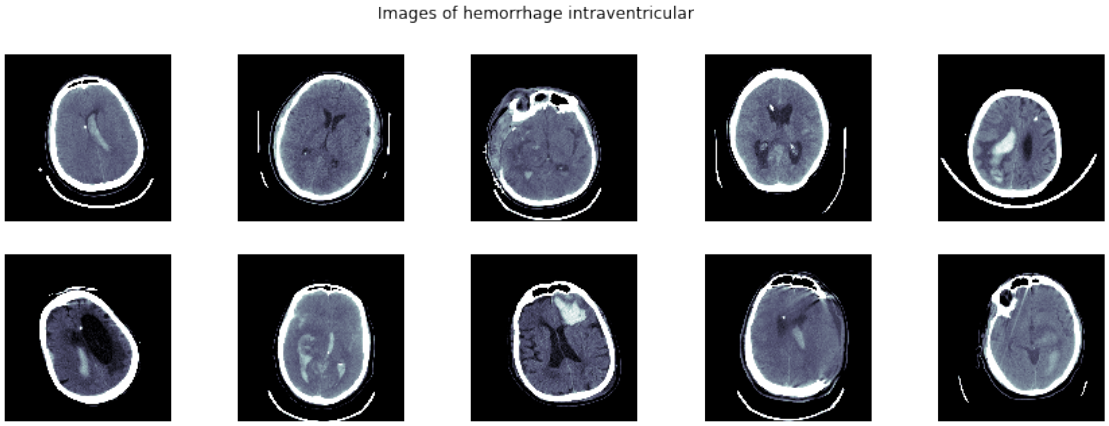



Figure 3.7

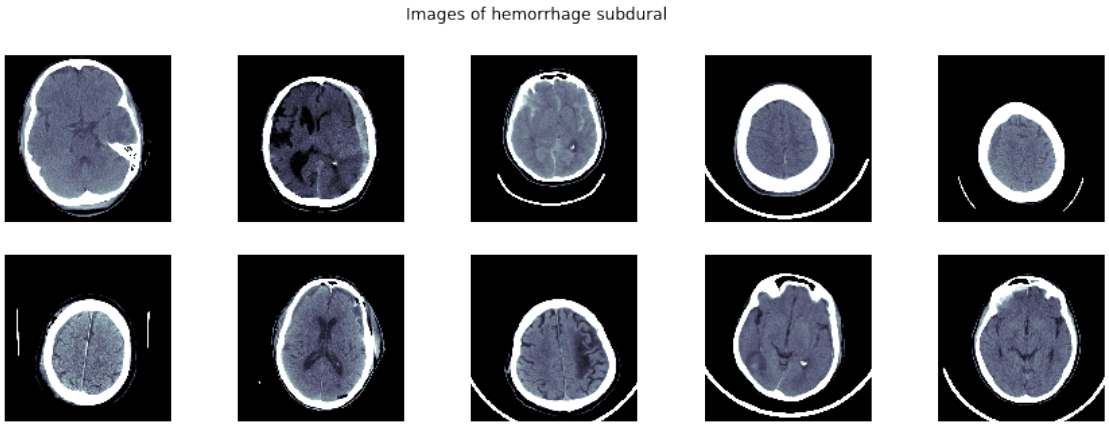



Figure 3.8

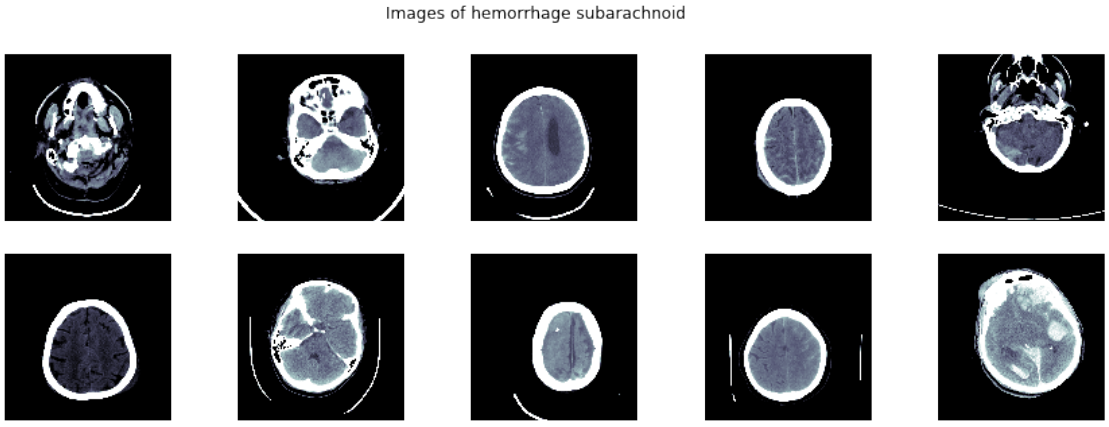



Figure 3.9

# Chapter 4

# Data Preprocessing

In any Machine Learning process, Data Preprocessing is that step in which the data gets transformed, or Encoded, to bring it to such a state that now the machine can easily parse it. In other words, the features of the data can now be easily interpreted by the algorithm.We have taken the reference from this website. [3]

## 4.1 CT Images

let us have a look on the CT image in Fig 3.1. We have seen that there value in the images have different range. We have to normalise the value.(shown in fig 4.1)

## 4.2 Conversion into Hounsefield Units

We have rescaled our image value into hounsefield unit. It is a simple linear transformation with the important values provided in the dicom header. We have taken idea from this website(shown in fig 4.2).

## 4.3 Windowing

Windowing, also known as grey-level mapping, contrast stretching, histogram modification or contrast enhancement is the process in which the CT image greyscale component of an image is manipulated via the CT numbers; doing this will change the appearance of the picture to highlight particular structures.The window width (WW) as the name suggests is the measure of the range of CT numbers that an image contains [10] .(shown in fig 4.3) There are at least 5 windows [2] that a radiologist goes through for each scan.

i)Brain Matter window : W:80 L:40
ii)Blood/subdural window: W:130-300 L:50-100
iii)Soft tissue window: W:350–400 L:20–60
iv)Bone window: W:2800 L:600
v)Grey-white differentiation window: W:8 L:32 or W:40 L:40

**Three window Scaling**:-We will stack image with three different window scaling Of (Brain + Subdural + Bone ). **One window Scaling**:-We will stack image with same window scaling Of (Brain + Brain + Brain

Think of a window as an instruction to the computer to highlight only voxels which filfill a specific value. L = window level or center W = window width or range

Example:
Brain Matter window
L = 40
W = 80
Voxels displayed range from 0 to 80
( Lower limit = 40 - (80/2), upper limit = 40 + (80/2) )

## 4.4 Resampling

A CT slice is typically reconstructed at 512 x 512 voxels, each slice represents approximately 370 mm of data in length and width. Using the metadata from the DICOM we can figure out the size of each voxel as the slice thickness. In order to display the CT in 3D isometric form (which we will do below), and also to compare between different scans, it would be useful to ensure that each slice is resampled in 1x1mm pixels and slices. [3] .(shown in fig 4.4)

## 4.5 Cropping

Cropping the images will help us to create better images and it will remove the unnecessary portion of the images. [3].(shown in fig 4.5)

## 4.6 Padding

Padding is used to equal the size of all images and it will also put brain in center.(shown in fig 4.6)

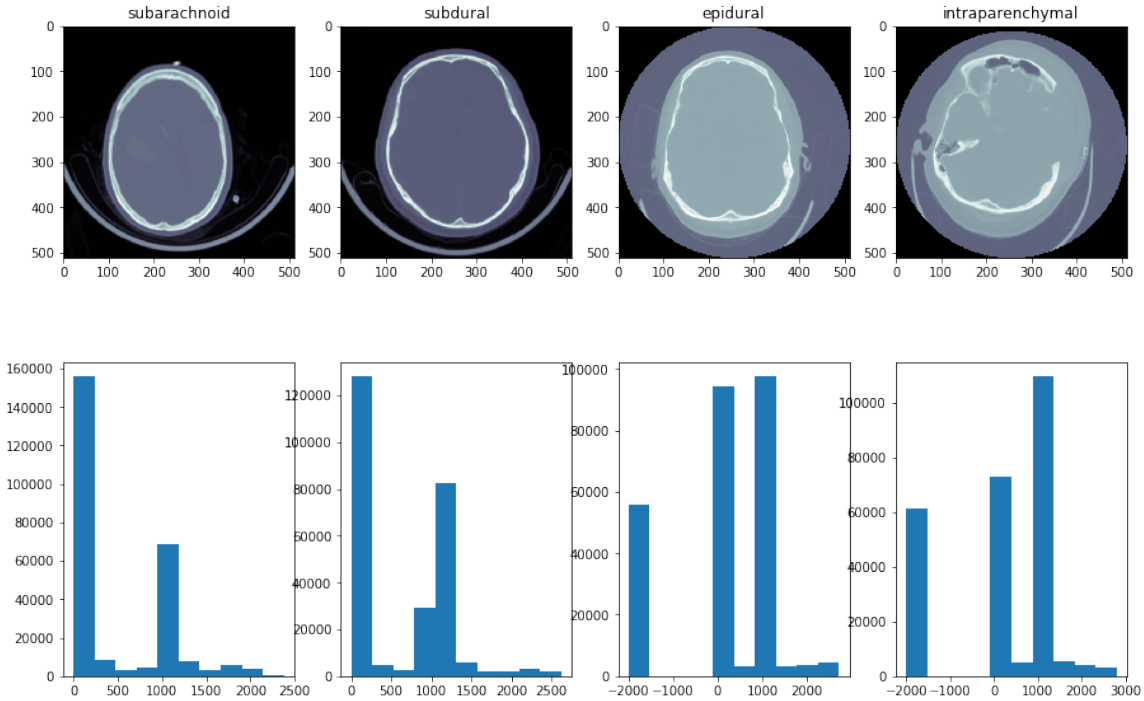



Figure 4.1: Visualisation of Ct images

## 4.7 Rescale

Normalising the values in between 0 and 256.(shown in fig 4.7)

## 4.8 Augment

Augmentation was important part of our preprocessing.We have horizontal flipping.

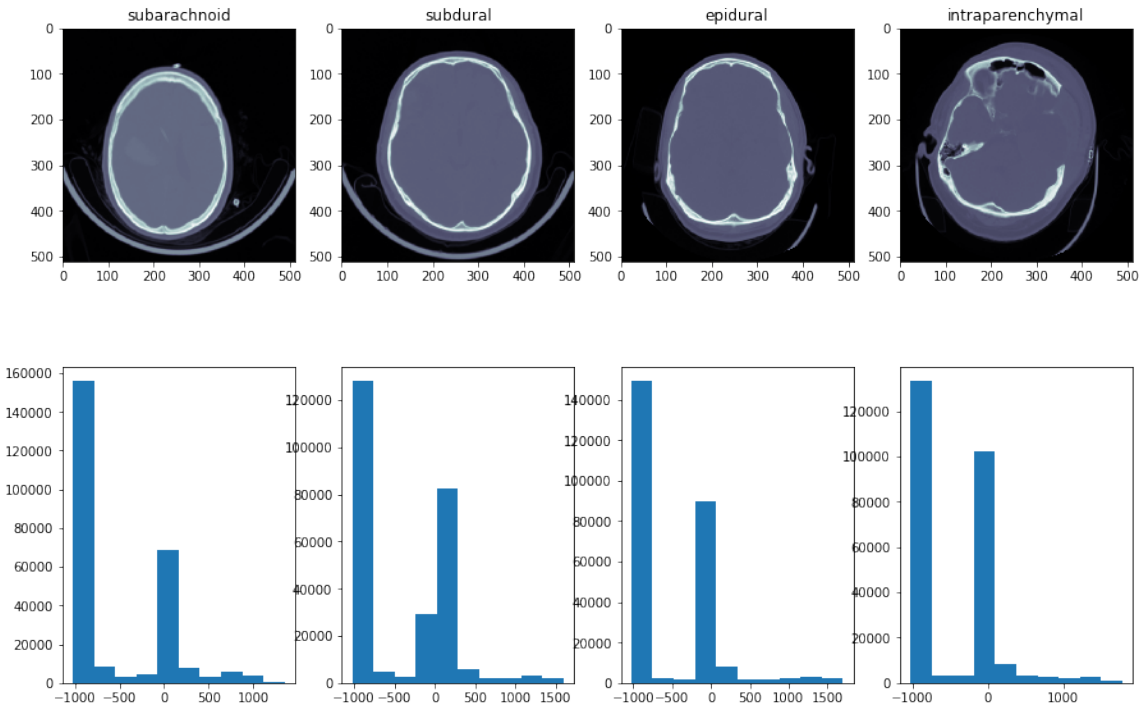



Figure 4.2: Rescaled Ct images into Hounsefield unit

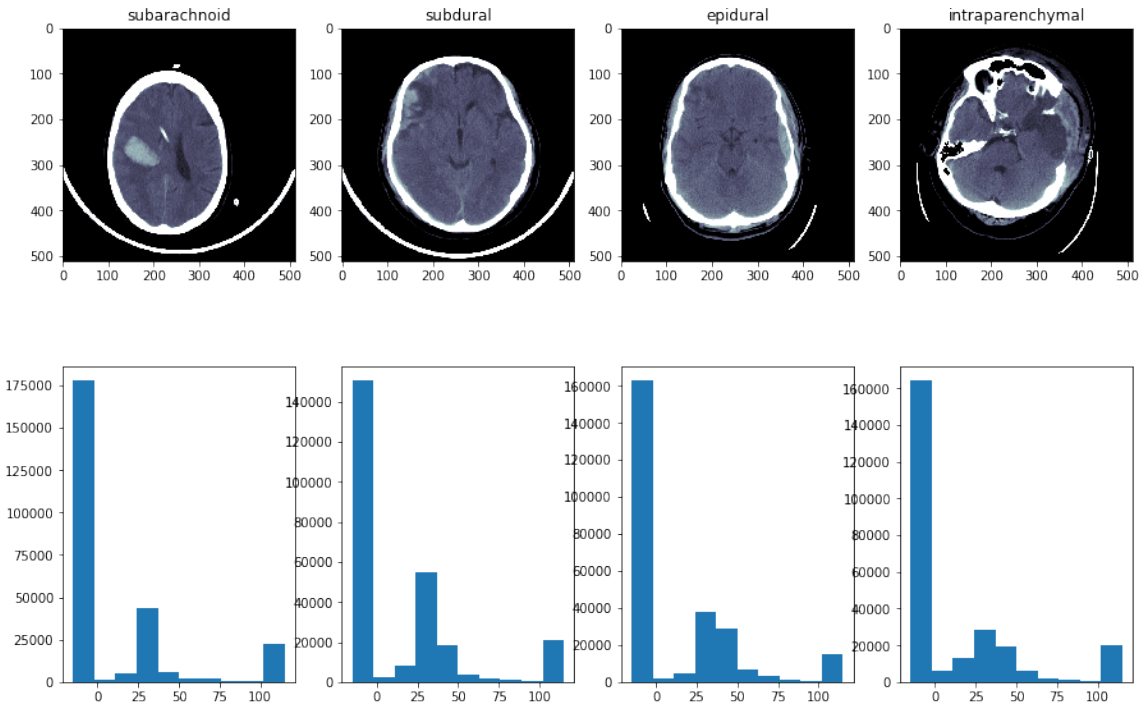



Figure 4.3: Image after window scaling

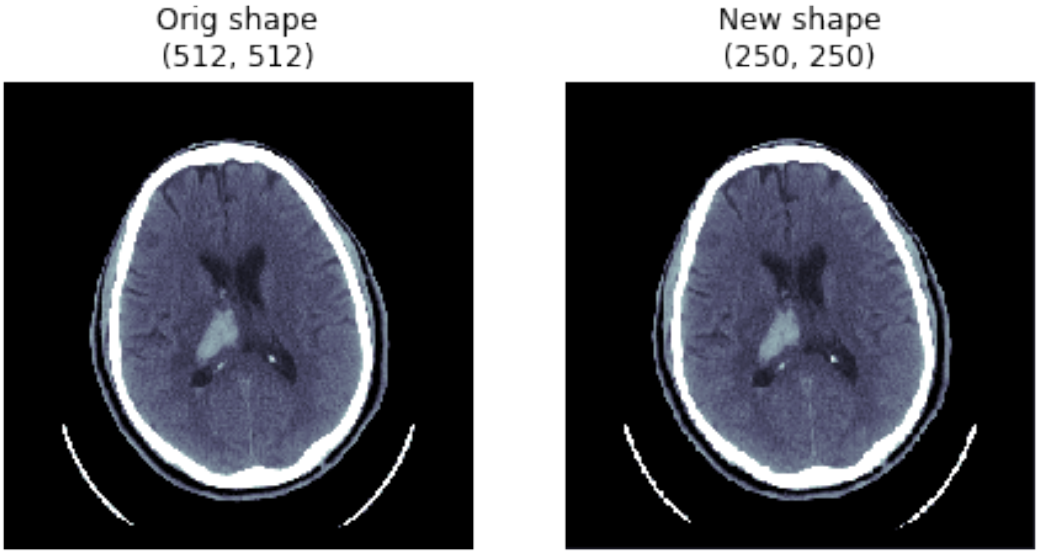



Figure 4.4: Image Resampling

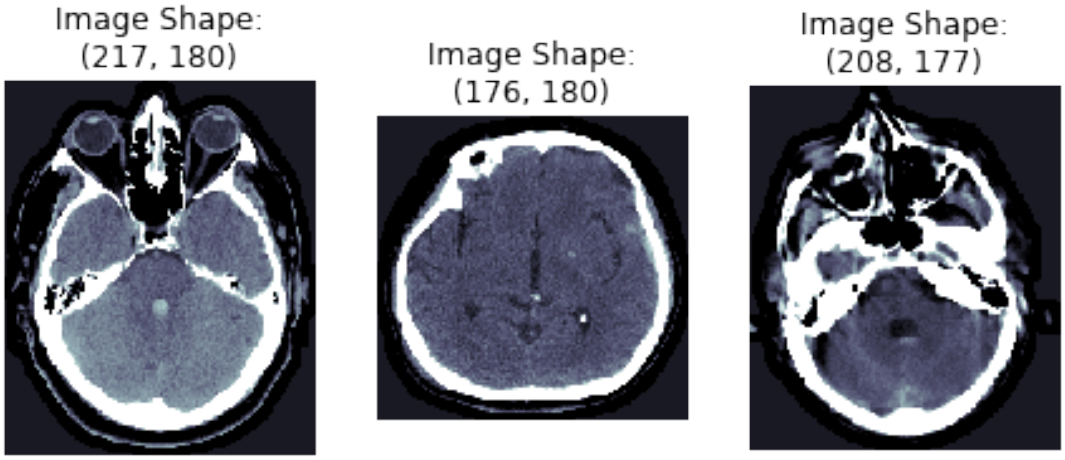



Figure 4.5: Some images after Cropping

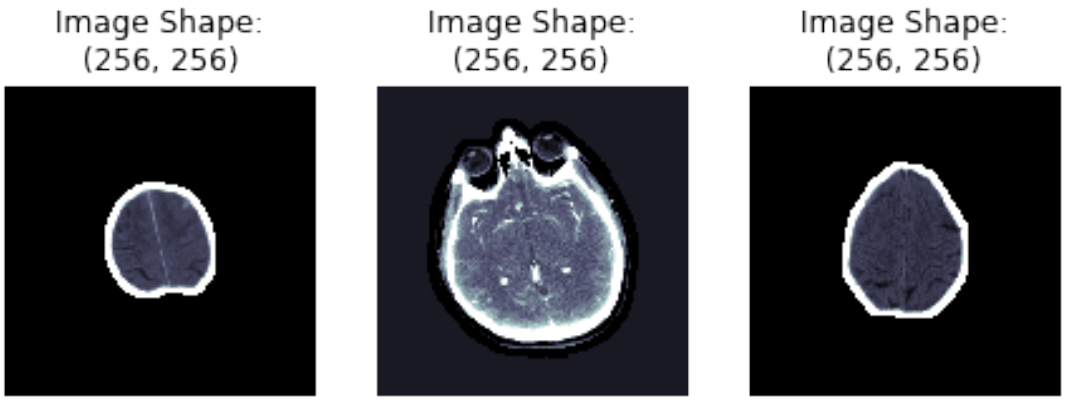



Figure 4.6: Some images after Padding

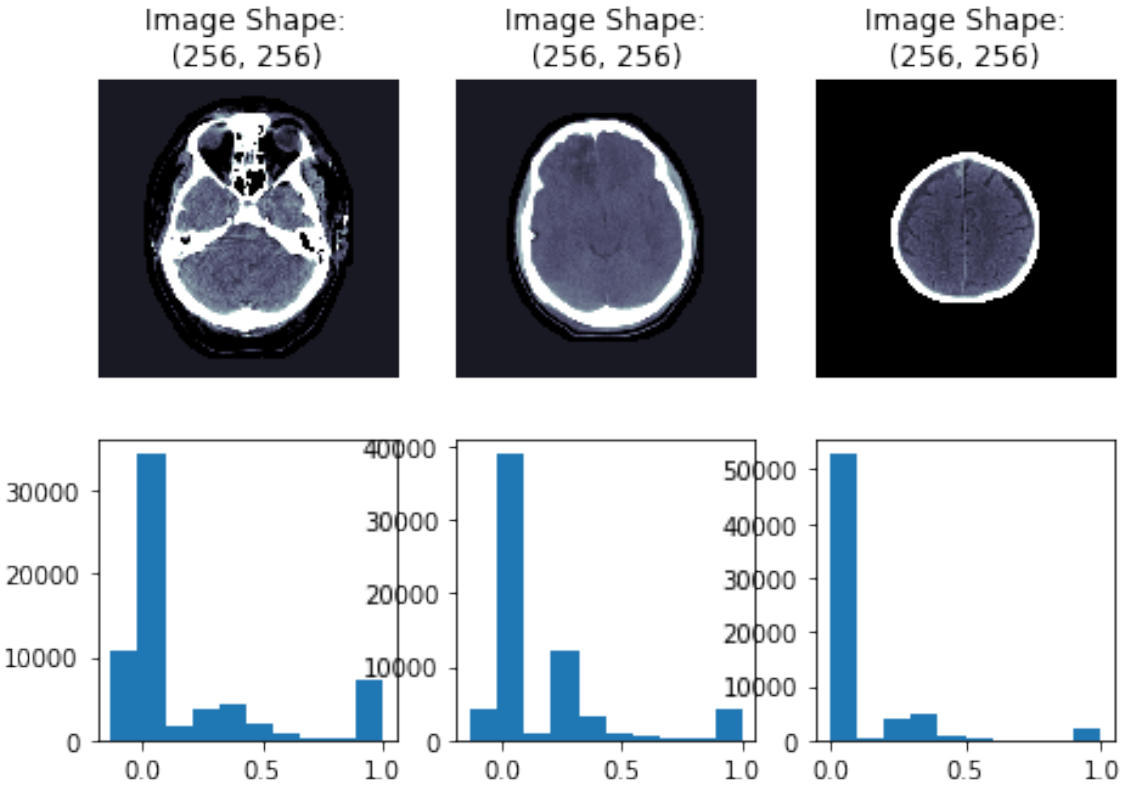



Figure 4.7: Images after Rescaling

# Chapter 5

# Multilabel Classification

## 5.1 Introduction

Multi-label classifications deal with multiple labels being assigned to every instance in a dataset. That is, an instance can be assigned more than one class simultaneously.It is concerned with learning a model that outputs a bipartition of a set of labels into relevant and irrelevant with respect to a query instance [20].In our problem statement we have to find a model that inputs the image of hemorrhage and it will output a vector y which will denote the probability of having each of the subtypes including any which determines whether any of the hemorrhage exists or not.

## 5.2 Loss function for Multilabel Classification

Binary crossentropy is a loss function that is used in binary classification tasks. These are tasks that answer a question with only two choices (yes or no, A or B, 0 or 1, left or right). Several independent such questions can be answered at the same time, as in multi-label classification . So in our problem to find the loss for each class using binary cross entropy whether that class is labelled as 1 or 0.So we find the loss of each class whether it is 1 or 0 and then we average the loss for all the class. If any of the class has wrong prediction then loss will increase and more the number of class wrong predicted then loss will increase Weighted Binary crossentropy is a loss function in which calculate loss using Binary crossentropy given weightage to each class.

## 5.3 Optimizing Function

Rectified Adam (RAdam), a novel variant of Adam, by introducing a term to rectify the variance the adaptive learning rate. [8] RAdam brings consistent improvement

over the vanilla Adam[5], which verifies the variance issue generally exists on various tasks across different network architectures. Convergence issue due to the undesirably large variance of the adaptive learning rate in the early stage of model training.

## 5.4 Activation Function

If each data point has exactly one output label, then the output node in fact should be depending on each other. We want the probability to sum to one over the output nodes.If the data point has multiple output labels in the sense that there are multiple tasks, we want to predict if a picture has an apple and to predict if a picture has a table. Then there isn't a need for the probability to sum to 1. So thats why we are using sigmoid as output function in the last layer.

## 5.5 Metrics for Multilabel Classification

**Precision**

: It calculates the proportion of positive predictions, those are actually correct.In our problem statement we are taking average of price of each class. [4]

**Recall**

: It calculates the proportion of actual positives that were identified correctly.In our problem statement we are taking average of recall of each class. [4]

**Weighted log loss**$_{m}etric$

:Logarithmic loss (related to cross-entropy) measures the performance of a classification model where the prediction input is a probability value between 0 and 1. The goal of our machine learning models is to minimize this value. A perfect model would have a log loss of 0. Log loss increases as the predicted probability diverges from the actual label.When we give weight to each of its class then its weight log loss.

# Chapter 6

# Intracranial Hemorrhage Detection using Resnet50

## 6.1 Motivation

We generally perceive that "the deeper the better " when it comes to convolutional neural network. This makes sense, since the models should be more capable (their flexibility to adapt to any space increases) because they have a bigger parameter space to explore. However, it has been noticed that after some depth, the performance degrades. This was one of the bottlenecks of earlier networks. ResNet gives us the residual learning framework to ease the training of networks that are substantially deeper than those used previously. It has won the 1st place on the ILSVRC 2015 for classification  localization task. [14] The depth of representation is very important for many visual recognition tasks and we have used this deep representation to generate class activation maps to indicate the discriminating image regions used by the CNN to identify that category .

## 6.2 Preliminaries

### 6.2.1 Architecture

Let us consider H(x) as an underlying mapping to be fit by a few stacked layers (not necessarily the entire net), with x denoting the inputs to the first of these layers. If one hypothesizes that multiple nonlinear layers can asymptotically approximate complicated functions2 , then it is equivalent to hypothesize that they can asymptotically approximate the residual functions, i.e., H(x)  x (assuming that the input and output are of the same dimensions). So rather than expect stacked layers to approximate H(x), we explicitly let these layers approximate a residual function F(x) := H(x)  x.

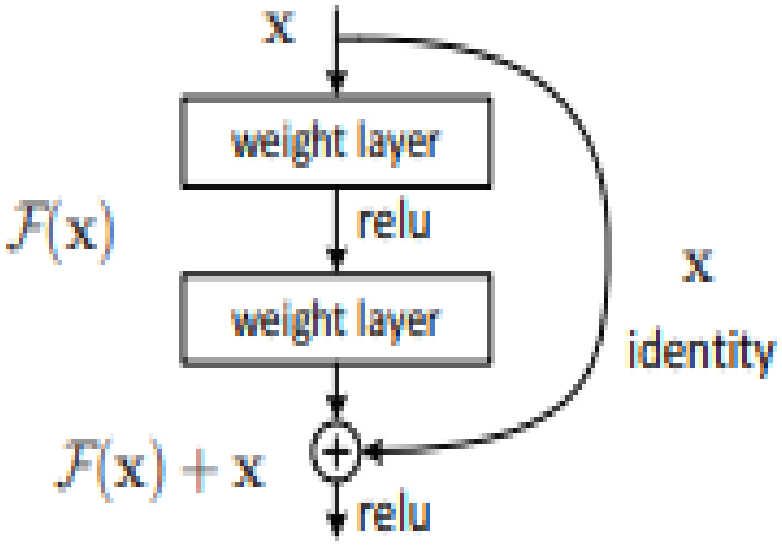



Figure 6.1: Residual learning: a building block. [14]

The original function thus becomes F(x)+x. Although both forms should be able to asymptotically approximate the desired functions (as hypothesized), the ease of learning might be different.

Based on the plain network fig we insert shortcut connections which turn the network into its counterpart residual version. The identity shortcuts can be directly used when the input and output are of the same dimensions. In fig we can see that the ResNet consists on one convolution and pooling step( on orange) followed by 4 layers of similar behavior. Each of the layers follow the same pattern. They perform 3 x 3 convolution with fixed feature map dimensions (F) [64, 128, 256, 512] respectively, by passing the input every 2 convolutions. Furthermore, the width (W) and Height (H) dimensions remain constant during the entire lay.

### 6.2.2 Implementations

We have used Dicom image of Intracranial Hemorrhage. We have rescaled that image value into hounsefield units. We have used same windows scale and width between 40 and 120 and stack the same image three times.We have resampled the images and crop the images to remove some unnecessary portion of the images.After that we have used padding to equal size of all the images and it will put the brain image into the center. We have used adaptive learning.We have trained our network Resnet50 from scratch and in the last layer we have added a full connected layer with six output

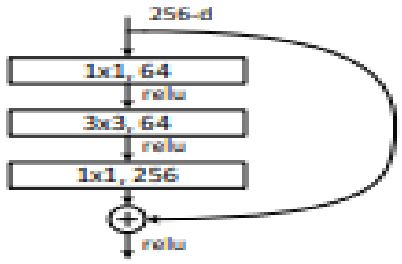



Figure 6.2: Building block for ResNet-50 [14]

node for six different class. We have used sigmoid as activation function in the last layer. We have used adam[12] as optimizing function and used BinaryCross entropy as loss function and used weighted log loss as metics. We have take our Batchsize of 32.We have trained it for 10 epochs and each epoch was taking 2 hours time.We have used some hyper parameter tunning of parameters like learning rate .

### 6.2.3 Result

We dont have the test label of our data and we usually get a score from our Kaggle after submission of prediction files. Kaggle website evaluated using a weighted multi-label logarithmic loss. Each hemorrhage sub-type is its own row for every image, and we are expected to predict a probability for that sub-type of hemorrhage. There is also an any label, which indicates that a hemorrhage of ANY kind exists in the image. The any label is weighted more highly than specific hemorrhage sub-types. We have used weighted log loss as metrics.We have given 2 weight to any and 1 to rest.Our training weighted log loss is 0.21 and validation weighted loss is 0.18. The score of Multilabel logarithmic loss that we get is 0.138.Our program rank was 367th in the competition.

# Chapter 7

# Intracranial Hemorrhage Detection using Mobile net v2

## 7.1 Motivation

A new mobile architecture, MobileNetV2 [16], that improves the state of the art performance of mobile models on multiple tasks and benchmarks as well as across a spectrum of different model sizes. We also describe efficient ways of applying these mobile models to object detection in a novel framework we call SSDLite. Additionally, we demonstrate how to build mobile semantic segmentation models through a reduced form of DeepLabv3 which we call Mobile DeepLabv3. is based on an inverted residual structure where the shortcut connections are between the thin bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. Additionally, we find that it is important to remove non-linearities in the narrow layers in order to maintain representational power. We demonstrate that this improves performance and provide an intuition that led to this design. [16]

## 7.2 Preliminaries

### 7.2.1 Architecture

Residual blocks(as shown in Figure 7.1) connect the beginning and end of a convolutional block with a skip connection. By adding these two states the network has the opportunity of accessing earlier activations that weren't modified in the convolutional block. This approach turned out to be essential in order to build networks of great depth.This idea as an inverted residual block because skip connections exist between narrow parts of the network which is opposite of how an original residual connection

| Input | Operator | $t$ | $c$ | $n$ | $s$ |
|---|---|---|---|---|---|
| $224^2 \times 3$ | conv2d | - | 32 | 1 | 2 |
| $112^2 \times 32$ | bottleneck | 1 | 16 | 1 | 1 |
| $112^2 \times 16$ | bottleneck | 6 | 24 | 2 | 2 |
| $56^2 \times 24$ | bottleneck | 6 | 32 | 3 | 2 |
| $28^2 \times 32$ | bottleneck | 6 | 64 | 4 | 2 |
| $14^2 \times 64$ | bottleneck | 6 | 96 | 3 | 1 |
| $14^2 \times 96$ | bottleneck | 6 | 160 | 3 | 2 |
| $7^2 \times 160$ | bottleneck | 6 | 320 | 1 | 1 |
| $7^2 \times 320$ | conv2d 1x1 | - | 1280 | 1 | 1 |
| $7^2 \times 1280$ | avgpool 7x7 | - | - | 1 | - |
| $1 \times 1 \times 1280$ | conv2d 1x1 | - | k | - | |

Figure 7.1: Bottleneck Architecture

works. When you run the two snippets above you will notice that the inverted block has far fewer parameters.

### 7.2.2 Implementations

We have used Dicom image of Hemorrhage. We have rescaled that image value into hounsefield units. We have used three different window scaling. We have resampled the images and crop the images to remove some unnecessary portion of the images . .After that we have used padding to equal size of all the images and it will put the brain image into the center. We have used adaptive learning.We have trained our network Mobilenetv2 from scratch and in the last layer we have added a full connected layer with six output node for six different class. We have used sigmoid as activation function in the last layer. We have used radam as optimizing function and used BinaryCross entropy as loss function and used weighted log loss as metics. We have take our Batchsize of 32.We have used here accuracy as a metrics also. We have trained it for 25 epochs and each epoch was taking an hour time. We have used some hyper parameter tunning of parameters like learning rate .

### 7.2.3 Result

We dont have the test label of our data.We usually get a score from Kaggle after submission of prediction files. Kaggle website evaluated using a weighted multi-label logarithmic loss. Each hemorrhage sub-type is its own row for every image, and we are expected to predict a probability for that sub-type of hemorrhage. There is also an any label, which indicates that a hemorrhage of ANY kind exists in the image. The any label is weighted more highly than specific hemorrhage sub-types.We have used weighted logloss as metrics.This model gives multi-label logarithmic loss value of 0.069.Our program was ranked 257 with this score in the competition.

# Chapter 8

# Intracranial Hemorrhage Detection using EfficientNetB2

## 8.1 Motivation

Scaling up ConvNets is widely used to achieve better accuracy. For example, ResNet [14] can be scaled up from ResNet-18 to ResNet-200 by using more layers; Rec4ently, GPipe [11] achieved 84.3 percent ImageNet top-1 accuracy by scaling up a baseline model four time larger. However, the process of scaling up ConvNets has never been well understood and there are currently many ways to do it. The most common way is to scale up ConvNets by their depth or width . Another less common, but increasingly popular, method is to scale up models by image resolution. In previous work, it is common to scale only one of the three dimensions – depth, width, and image size. Though it is possible to scale two or three dimensions arbitrarily, arbitrary scaling requires tedious manual tuning and still often yields sub-optimal accuracy and efficiency. In this paper, we want to study and rethink the process of scaling up ConvNets. [17]

## 8.2 Preliminaries

### 8.2.1 Architecture

There are three scaling dimensions of a CNN: depth, width, and resolution. Depth simply means how deep the networks is which is equivalent to the number of layers in it. Width means how wide the network is. One measure of width, for example, is the number of channels in a Conv layer whereas Resolution is simply the image resolution that is being passed to a CNN. The figure below 8.1 will give you a clear idea of what scaling means across different dimensions. We will discuss these in detail as well. [6]

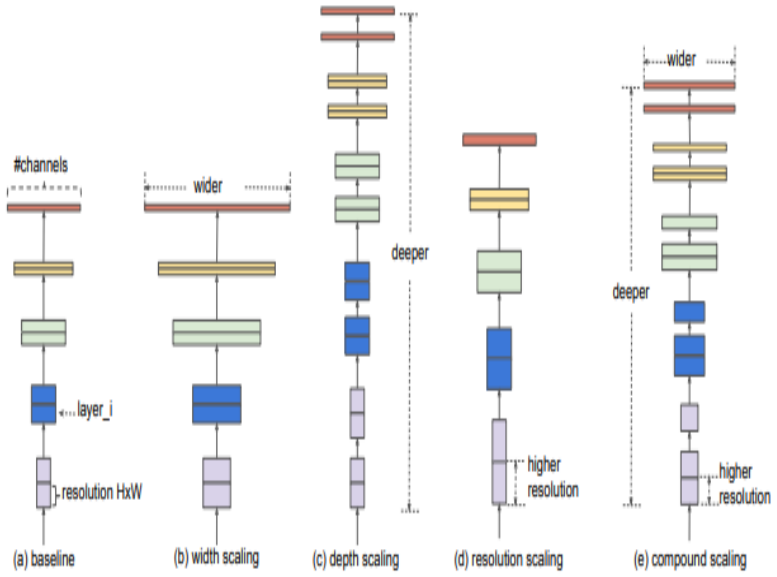



Figure 8.1: Model Scaling. [17]

**EfficientNet** Scaling doesn't change the layer operations, hence it is better to first have a good baseline network and then scale it along different dimensions using the proposed compound scaling. The authors obtained their base network by doing a Neural Architecture Search (NAS) that optimizes for both accuracy and FLOPS. The architecture is similar to M-NASNet as it has been found using the similar search space [6]. The network layers/blocks are as shown below:

### 8.2.2 Implementations

We have used Dicom image of Hemorrhage. We have rescaled that image value into hounsefield units. We have used three different window scaling. We have resampled the images and crop the images to remove some unnecessary portion of the images.After that we have used padding to equal size of all the images and it will put the brain image into the center. We have used adaptive learning.We have trained our network EfficientNetB2 from scratch and in the last layer we have added a full connected layer with six output node for six different class. We have used sigmoid as activation function in the last layer. We have used radam as optimizing function and used BinaryCross entropy as loss function .We have take our Batchsize of 32. We have used here precise and recall as a metrics also.We have trained it for 25 epochs and each epoch was taking 2 hours time. We have used some hyper parameter tunning of parameters like learning rate .

| Stage $i$ | Operator $\hat{\mathcal{F}}_i$ | Resolution $\hat{H}_i \times \hat{W}_i$ | #Channels $\hat{C}_i$ | #Layers $\hat{L}_i$ |
|---|---|---|---|---|
| 1 | Conv3x3 | 224 × 224 | 32 | 1 |
| 2 | MBConv1, k3x3 | 112 × 112 | 16 | 1 |
| 3 | MBConv6, k3x3 | 112 × 112 | 24 | 2 |
| 4 | MBConv6, k5x5 | 56 × 56 | 40 | 2 |
| 5 | MBConv6, k3x3 | 28 × 28 | 80 | 3 |
| 6 | MBConv6, k5x5 | 28 × 28 | 112 | 3 |
| 7 | MBConv6, k5x5 | 14 × 14 | 192 | 4 |
| 8 | MBConv6, k3x3 | 7 × 7 | 320 | 1 |
| 9 | Conv1x1 & Pooling & FC | 7 × 7 | 1280 | 1 |

Figure 8.2: Architecture

### 8.2.3 Result

Since we dont have the test label of our data. We usually get a score from Kaggle after submission of prediction files. Kaggle website evaluated using a weighted multi-label logarithmic loss. Each hemorrhage sub-type is its own row for every image, and you are expected to predict a probability for that sub-type of hemorrhage.There is also an any label, which indicates that a hemorrhage of ANY kind exists in the image. The any label is weighted more highly than specific hemorrhage sub-types. This model gives a training precise is 0.893 and recall is 0.8323 and validation precise is 0.893 and recall is 0.8323. This model gives multi-label logarithmic loss value of 0.064. Our program was ranked 195 with these score in the competition.

# Chapter 9

# Future Work and Conclusion

## 9.1 Conclusion

In these project we have achieved a multilabel logorithmic loss value of 0.064. We have started with our base line model resnet which has given a score 0.0138. In resnet model we have used single window scaling. But after that we have used three window scaling of dicom images and trained for 25 epochs that gives a better result and our score value was 0.069 and after that we have used EfficientnetB2 that has also given better score 0.064. Our intial model rank on kaggle leaderboard was 367 and our final score gives a rank aroud 195. Most of the Kaggle top rankers have converted the 2D slice into 3D then used that data but our objective was to classify from a single slide .

## 9.2 Future Work

Weighted Binary Cross Entropy can be used as a loss function. Since we have limited GPU access we were unable to tune some of the hyperparamaters.

# Bibliography


[1] Subdural hemorrhage https://www.aurorahealthcare.org/services/neuroscience/brain-skull-base-care/brain-hemorrhage/subdural-hemorrhage

[2] Kaggle RSNA Intracranial Hemorrhage Detection (2019)

[3] Andrew Murphy: Dicom processing and segmentation in python https://www.raddq.com/dicom-processing-segmentation-visualization-in-python/

[4] Brownlee, J.: Calculate precision, recall, https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/

[5] Diederik P. Kingma , Jimmy Ba: Adam: A method for stochastic optimization (2014)

[6] Engineer, M.T.S.S.: Efficientnet: Improving accuracy and efficiency through automl and model scaling

[7] Freeman, William David, a.: Intracranial hemorrhage: diagnosis and management. Neurologic clinics vol. 30, (2012), doi:10.1016/j.ncl.2011.09.002

[8] Liyuan Liu and Haoming Jiang, He, P., Chen, W., Liu, X., Gao, J., Han, J.: On the variance of the adaptive learning rate and beyond (2019)

[9] Hinson, H. E., H.D.F..Z.W.C.: Management of intraventricular hemorrhage https://doi.org/10.1007/s11910-010-0086-6

[10] Howard Chen: Windowing, https://www.raddq.com/dicom-processing-segmentation-visualization-in-python/

[11] Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, M.X., Chen, D., Lee, H., Ngiam, J., Le, Q.V., Wu, Y., Chen, Z.: Gpipe: Efficient training of giant neural networks using pipeline parallelism (2018)

[12] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International

Conference on International Conference on Machine Learning - Volume 37. p. 448–456. ICML'15, JMLR.org (2015)

[13] Mayfield ClinicAuthor links open overlay panelMichael K.AbrahamMD, M.T.W.: Emergency medicine clinics of north america https://doi.org/10.1016/j.emc.2016.06.011

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun: Deep residual learning for image recognition (2015)

[15] Kuo, W., Hne, C., Mukherjee, P., Malik, J., Yuh, E.L.: Expert-level detection of acute intracranial year = 2019, url = 10.1073/pnas.1908021116, publisher = National Academy of Sciences,

[16] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh : Mobilenetv2: Inverted residuals and linear bottlenecks (2018)

[17] Mingxing Tan, Quoc V. Le: Efficientnet: Rethinking model scaling for convolutional neural networks (2019)

[18] Mohammad R. Arbabshirani, Brandon K. Fornwalt, G.J.M.J.D.S.B.D.G.A.A.P., Moore, G.J.: Advanced machine learning in action: identification of intracranial hemorrhage on computed tomography scans of the head with clinical workflow integration https://doi.org/10.1038/s41746-017-0015-z

[19] Pryse-Phillips, W.: Companion to Clinical Neurology

[20] Tawiah, C.A., Sheng, V.S.: A study on multi-label classification. In: Perner, P. (ed.) Advances in Data Mining. Applications and Theoretical Aspects. pp. 137–150. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)

[21] wikipedia: Intraparenchymal$_h emorrhage$