A
Major Project Report

On

"PREDICTING MENINGIOMA USING INCEPTION AND EFFICIENTNET ALGORITHMS"

Submitted in partial fulfillment of the

Requirement for the award of the degree of

Bachelor of Technology

In

Computer Science & Engineering

By

T.VISHAL - 18R21A05B6
P.HEMANTH GOUD - 18R21A05A6
K.BHARGAV - 18R21A0579
T.ROHIT SRIVATSA -15R21A05P2

Under the guidance of

Mrs. P. Devika
Assistant Professor

Department of Computer Science & Engineering

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS

MLR INSTITUTE OF TECHNOLOGY
(UGC AUTONOMOUS)
Affiliated to JNTUH,Approved by AICTE
Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

NATIONAL BOARD ACCREDITATION

ACCREDITED WITH GRADE A NAAC

2018-2022

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS

MLR INSTITUTE OF TECHNOLOGY
(UGC AUTONOMOUS)
Affiliated to JNTUH,Approved by AICTE
Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

Department of Computer Science & Engineering

CERTIFICATE

This is to certify that the project entitled "**PREDICTING MENINGIOMA USING INCEPTION AND EFFICIENTNET ALGORITHMS"** has been submitted by **T.VISHAL(18R21A05B6), P.HEMANTHGOUD(18R21A05A6), K.BHARGAV(18R21A0579), T.ROHIT SRIVASTA(15R21A05P2)** in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering from Jawaharlal Nehru Technological University, Hyderabad. The results embodied in this project have not been submitted to any other University or Institution for the award of any degree or diploma.

Internal Guide	**Head of the Department**
Mrs. P Devika	**Dr. E Anupriya**
Assistant Professor	Professor

External Examiner

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS MLR INSTITUTE OF TECHNOLOGY

(UGC AUTONOMOUS)

Affiliated to JNTUH, Approved by AICTE

Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

NATIONAL BOARD ACCREDITATION

ACCREDITED WITH GRADE A NAAC

Department of Computer Science & Engineering

DECLARATION

We hereby declare that the project entitled **"PREDICTING MENINGIOMA USING INCEPTION AND EFFICIENTNET ALGORITHMS"** is the work done during the period from **January 2022 to June 2022** and is submitted in partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering from Jawaharlal Nehru Technological University, Hyderabad. The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

T.VISHAL - 18R21A05B6

P.HEMANTH GOUD - 18R21A05A6

K.BHARGAV - 18R21A0579

T.ROHIT SRIVATSA - 15R21A05P2

MARRI LAXMAN REDDY GROUP OF INSTITUTIONS

MLR INSTITUTE OF TECHNOLOGY

(UGC AUTONOMOUS)

Affiliated to JNTUH,Approved by AICTE

Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

NATIONAL BOARD ACCREDITATION

ACCREDITED WITH GRADE A NAAC

Department of Computer Science & Engineering

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that we now have the opportunity to express our guidance for all of them.

First of all We would like to express our deep gratitude towards our internal guide **Mrs. P. Devika, Assistant Professor, Department of CSE** for his support in the completion of our dissertation. We wish to express our sincere thanks to **Dr. E Anupriya, HOD, Dept. of CSE** and also **principal Dr. K. SRINIVAS RAO** for providing the facilities to complete the dissertation.

We would like to thank all our faculty and friends for their help and constructive criticism during the project period. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

T.VISHAL - 18R21A05B6
P.HEMANTH GOUD - 18R21A05A6
K.BHARGAV - 18R21A0579
T.ROHIT SRIVATSA - 15R21A05P2

Department of Computer Science & Engineering

ABSTRACT

Meningioma is the most common primary brain tumor, accounting for more than 30% of all brain tumors. Meningiomas originate in the meninges, the outer three layers of tissue that cover and protect the brain just under the skull. Women are diagnosed with meningiomas more often than men. About 85% of meningiomas are noncancerous, slow-growing tumors. Almost all meningiomas are considered benign, but some meningiomas can be persistent and come back after treatment.

This LGG is treated with a combination of surgery and examination by monitoring the meningioma with brain MRI scans. The study of this project enabled us to build a fully automated system for segmentation of meningioma utilizing computer vision techniques, and developing models that would allow high-quality LGG identification in the brain MRI would potentially be automated to identify the genomic subtype of the meningioma by rapid and inexpensive imaging. The methods, techniques, advantages and their limitations and their future challenges are discussed in this project.

Keywords—glioma, LGG, MRI,Inception,EfficientNet

LIST OF FIGURES

INDEX

CHAPTER 1

1. INTRODUCTION

1.1 Overview

Brain tumors are one of the most common nervous system illnesses that do the most harm to human health. Glioma is one of the most lethal and deadly intracranial tumors. The patients who developed HGG have an estimated life expectancy of roughly two years. This is usually divided into two types: high-grade glioma (HGG) and low-grade glioma (LGG) (LGG). MRIs, Computed Tomography (CT), and Tomography with Positron Emission have all been used to analyze brain cancers. MRI imaging technique was found to be significant for accurately diagnosing brain tumors out of all imaging modalities.

A current cancer study dis- covered a new study known as radio-genomics. These cutting-edge medical procedures aid in the accurate research of malignancies. The advancement of artificial intelligence techniques works together to solve these tumour identifications.

Expert systems have a wide range of effective applications, particularly in the medical arena. These expert algorithms have even outperformed professional radiologists in several cases.

Deep learning-based automatic segmentation approaches have yielded considerable results in recent years. Convolutional Neural Networks are the most renowned and verified deep learning models for extracting characteristics that are advantageous in real-time classification given the original data. Given its established outcomes and accomplishments, the medical brain tumor segmentation shifted its focus to deep learning.

Recently, breakthroughs in multi-scale feature extraction methods have demonstrated the segmentation job. Although there are several proposed methods for extracting features and creating feature maps for segmentation, we present a new network in this white paper that combines residual blocks, commonly known as skip connections, with the pre-existing Unet architecture proposed by Olaf Ronneberger and team.

1.2 Purpose of the project

CNN models are progressively mind-boggling, for certain frameworks having in excess of 100 layers, which implies a great many loads and billions of associations between neurons. A common CNN design contains ensuing layers of convolution, pooling, actuation, and order (completely associated). The convolutional layer includes maps by convolving a bit across the information picture. The pooling layer is utilized to downsample the yield of going before convolutional layers by utilizing the most extreme or normal of the characterized neighborhood as the worth passed to the following layer. Redressed Linear Unit (ReLU) and its changes, for example, Leaky ReLU are among the most generally utilized enactment capacities. ReLU nonlinearly changes information by cutting any negative information esteems to nothing while positive info esteems are passed as yield. To play out an expectation of information, the yield scores of the last CNN layer are associated with misfortune work (e.g., a cross-entropy misfortune that standardizes scores into multinomial dispersion over names). At last, boundaries of the organization are found by limiting a misfortune work among forecast and ground truth names with regularization limitations, and the organization loads are refreshed at every emphasis (e.g., utilizing stochastic slope drop – SGD) utilizing backpropagation until assembly.

1.3 Motivation

As an essential errand in Computer vision, the semantic division can give principal data to protest location and example division to help man-made consciousness better comprehend this present reality. Since the proposition of completely convolutional neural organization (CNN), it has been broadly utilized in the semantic division as a result of its high exactness of pixel-wise grouping just as high accuracy of restriction. In this paper, we apply a few popular FCNN to mind tumor division, making examinations and changing organizational structures to accomplish better execution estimated by measurements, for example, exactness, review, mean of convergence of association (MoU), and dice score coefficient (DSC). The acclimations to the exemplary FCNN incorporate adding more associations between convolutional layers, broadening decoders after up example layers, and changing the way shallower layers' data is reused. Other than the construction alteration, we likewise propose another classifier with a progressive dice misfortune. Propelled by the containing connection between classes, the misfortune work changes numerous groupings over to different double arrangements to neutralize the negative impact brought about by awkwardness informational index. Huge analyses have been done on the preparation set and testing set to evaluate our refined completely convolutional neural organizations and new kinds of misfortune work. Serious figures demonstrate they are more successful than their archetypes.

Deep learning has demonstrated as of late to be a useful asset for picture examination and is currently broadly used to section both 2D and 3D clinical pictures. Profound learning division systems depend on the decision of organization engineering as well as on the decision of misfortune work. To alleviate this issue, systems, for example, the weighted cross-entropy work, the affectability work, or the Dice misfortune work, have been proposed. In this work, we explore the conduct of these misfortune capacities and their affectability to learning rate tuning within the sight of various places of name irregularity across 2D and 3D division undertakings. We additionally propose to utilize the class re-adjusting properties of the Generalized Dice cover, a known measurement for division evaluation, as a powerful and exact profound learning misfortune work for lopsided assignments.

CHAPTER 2

2. LITERATURE SURVEY

2.1 Existing System

Second rate gliomas are normally treated with a mix of a medical procedure, perception, and radiation. In the event that the tumor is situated in a zone where it is protected to eliminate, at that point the neurosurgeon will endeavor to eliminate however much as could be expected. Now and then this is all the treatment you will require toward the start and your primary care physicians will screen your tumor with MRI checks at regular intervals. In the event that the tumor seems, by all accounts, to be developing, your PCPs will at that point consider either performing another medical procedure or beginning therapy with radiation. There is no demonstrated job for chemotherapy in treating poor quality gliomas, however when the tumor develops regardless of radiation and chemotherapy, your primary care physicians may choose to utilize it.

Regardless of whether the whole noticeable tumor is taken out at a medical procedure, there are typically some tumor cells that have attacked adjoining portions of the cerebrum. These cells can develop and make the tumor return. Ultimately, most second rate gliomas will proceed to develop and afterward form into a higher evaluation tumor, for example, the evaluation 3 or evaluation 4 tumors.

Determination of LGGs is made through a blend of imaging, histopathology, and sub-atomic demonstrative strategies. On a registered tomography filter, second rate gliomas show up as diffuse zones of low constriction. On customary attractive reverberation imaging (MRI), which is as of now the imaging methodology of decision, LGGs are frequently homogeneous with low sign force on T1-weighted successions and hyperintensity on T2-weighted and Fluid-Attenuated Inversion Recovery (FLAIR) groupings . Calcifications might be obvious as regions of T2 hyperintensity/T1 hypointensity in up to 20% of injuries, including oligodendrogliomas and astrocytomas, and are especially reminiscent of oligodendrogliomas. Gliomas, all in all, invade the encompassing parenchyma in spite of obvious radiographic edges seen on T2/FLAIR arrangements. Differentiation improvement, if present, is insignificant, and is bound to be seen with oligodendrogliomas. In spite of the fact that contrast improvement has been traditionally connected with a more significant level of danger, some level of differentiation upgrade might be seen in up to 60% of LGG. LGGs vary from grade III and IV gliomas, as the last frequently exhibit a more significant level of tumor heterogeneity and difference upgrade, confined dispersion on dissemination weighted imaging attractive reverberation (MR) groupings, and expanded relative cerebral blood volume on perfusion-weighted MRI. Regardless of trademark radiographic discoveries, tumor grade can't be dictated by imaging alone. More up to date imaging strategies, for example, MR spectroscopy (MRS) and positron discharge tomography (PET) imaging, may improve the demonstrative potential; notwithstanding, right now, histopathologic assessment of tissue stays the best quality level for conclusion and reviewing of LGG.

The fundamental objective of medical procedure is to acquire obsessive finding and, when plausible, to accomplish a gross absolute resection. Advances, for example, preoperative utilitarian MRI and tractography, just as intraoperative neurophysiological checking, permit specialists to securely expand resection of T2/FLAIR anomalies on MRI frequently including persuasive regions. Patients with tumors that can't be securely resected, or who have injuries of dubious etiology, may go through stereotactic biopsy utilizing preoperative or intraoperative MRI imaging to acquire tissue for histopathological investigation. Specialists focus on the possibly higher evaluation segment of the sore (for instance, contrast improvement) for biopsy. The yield of such biopsies is as high as 90%–95%; notwithstanding, due to the likely heterogeneity of these tumors, biopsy may not mirror the most elevated evaluation for analysis, with revealed exactness rates going from 51% to 83%.

Histopathology

The tissue test is stained utilizing hematoxylin and eosin, which takes into consideration distinguishing proof and arrangement of tumor type. Diffuse astrocytomas consist of very much separated fibrillary or gemistocytes neoplastic astrocytes on a free network. Oligoastrocytomas are diffusely penetrating tumors with a combination of oligodendroglial and astrocytic cell types. Oligodendrogliomas are invading tumors containing cells with uniform-showing up cores and perinuclear clearing, frequently portrayed as having a "singed egg" appearance.

Progressively, considers have upheld careful resection as opposed to perception to improve generally endurance. Also, a few examinations recommend an advantage of degree of resection on movement free endurance. Regardless of whether gliomas are by chance found or suggestive, medical procedure has been accounted for to improve seizure control.

Surgery

In one survey of the careful administration of LGG, the creators noticed the authentic contentions for attentive holding up in chose patients with insignificant or medicinally controlled indications, with one of the essential contentions dependent on information proposing that such a methodology didn't deteriorate patients' QoL, nor did it adversely affect generally speaking endurance, albeit the estimation of such information is restricted by its review nature. Of nine review careful investigations, six exhibited huge generally speaking endurance advantage with broad careful resection. Two forthcoming preliminaries assessing resection and postoperative radiation treatment exhibited a huge endurance advantage with more forceful resection on univariate investigation, yet not on multivariate examination. These examinations are restricted by unblinded evaluation of resection (i.e., much of the time, the specialist decides the degree of resection), just as patient and treatment determination inclinations. In another audit, the creators analyzed all significant distributions since 1990 tending with the impact of degree of careful resection on glioma result. They reasoned that there was a pattern toward progress in endurance with more broad careful resection. In univariate and multivariate investigations of these LGG examinations, they noticed that degree of resection had huge prognostic incentive in 7 of the 10 examinations.

Radiation

A few forthcoming clinical preliminaries have analyzed the utility of high-portion versus low-portion radiation and the expenses versus advantages of early versus postponed radiotherapy. In EORTC 22844, examiners evaluated the general adequacy of radiotherapy and the capability of a portion reaction relationship. An aggregate of 379 grown-up patients with LGG was randomized to get radiotherapy postoperatively (or postbiopsy) with 45 Gy in 5 weeks versus 59.4 Gy in 6.6 weeks. At a middle development of 74 months, there was no huge contrast in by and large endurance (58% in the low-portion gathering and 59% in the high-portion gathering) or movement free endurance (47% in the low-portion gathering and half in the high-portion gathering), and there was no obvious portion reaction relationship for radiotherapy in LGG.

Comparable outcomes were seen in the North Central Cancer Treatment Group/Radiation Therapy Oncology Group (RTOG)/Eastern Cooperative Oncology Group (ECOG) study, in which endurance and harmfulness were assessed in low-and high-portion radiation arms. In this examination, 203 patients with supratentorial LGG from 1986 to 1994 were randomized to either the low-portion (50.4 Gy in 28 parts) or high-portion (64.8 Gy in 36 divisions) treatment gathering. There was no bit of leeway of higher-portion radiation treatment saw in this examination. Truth be told, there was a pattern toward improved endurance at 2 and 5 years with low-portion treatment (2-and 5-year endurance of 94% and 72%, separately, in examination with 85% and 64% with high-portion treatment). Notwithstanding, this distinction didn't arrive at measurable importance. Likewise, there was a higher frequency of radiation neurotoxicity (radiation putrefaction) in the high-portion radiotherapy gathering (5% versus 2.5% in the low-portion radiotherapy gathering).

Chemotherapy

In spite of the fact that chemotherapy is frequently utilized in high-grade gliomas, its part, with or without radiation treatment, in the therapy of patients with LGG stays a subject of examination. Given its exhibited viability in the high-grade glioma populace, temozolomide has been specifically noteworthy. In one stage II investigation, there was exhibited movement of temozolomide in LGG patients with reformist sickness. In this accomplice of 46 patients, 61% of subjects accomplished radiographic reactions—24% having accomplished total reaction and 37% having accomplished fractional reaction. The middle movement free endurance (PFS) was 22 months, with a 6-month PFS of 98% and a year PFS of 76%.

Monitoring Response to Treatment

The ideal technique for surveying treatment reactions in LGG stays a functioning region of examination. Presently, MRI (T2/FLAIR succession), with or without contrast improvement, is utilized to recognize tumor size and related peritumoral edema. A few creators recommend that treatment results may be all the more dependably assessed utilizing progressed imaging methods intended to evaluate explicit organic parts of the tumor, including amino corrosive PET, MRS, or potentially cerebral blood volume appraisal with perfusion-weighted MRI. Be that as it may, none of these elective imaging markers have been approved for use in LGG clinical preliminaries or in clinical practice.

CHAPTER 3

3. PROPOSED SYSTEM

3.1 Proposed System

We approached this project to learn more about the DEEP NEURAL NETWORKS not merely to sneak the code from tutorials and use correct syntaxes to build something. With this in mind, first we developed a lot of programs using Python, CNN modules with both the high level and the low level programming in essence, using the Keras and Tensorflow libraries which implements different aspects of the neural network ability and which were totally related to our project. Among those implementations of the Sequential model, transfer learning of the bottleneck features gave clear conclusions and documentation also helped a lot. Now, after getting pretty much exposure in model development, we decided to go ahead with our project. Ignoring the futile attempts to add many features in our model, which fortunately could not be made possible, the work of our final model can be divided into following phases:

Input Image:

The input image taken from the kaggle dataset was in .tif format. Although the image format is not in JPEG or PNG, the format can be directly used for training without any conversions. Also, the size of the image was 256*256, which is a minimized and preferred size for most deep learning models.

Pre-processing:

It is used to improve the quality of the digital image, especially poor image quality such as images with too much noise, specifically blurriness. To remove the noise we use Keras library, which helps in preprocessing the image based on the model we choose.

Skull Stripping:

It is used to remove the unwanted or the unnecessary regions of the MRI image. It helps in identifying the pattern of the tumor while training the digital images.

Detection Process:

In the detection process, to detect the tumor of any given bio-medical image, we have used the transfer learning technique to classify the image. ResNet50 and VGG16, both were used during the model training.

Classification:

Abnormality of an image is determined by classification in essence abnormal or normal. There are many classifiers but the neural network is one of the most prominent classifiers that differentiate between images affected with tumors or normal medical images. There will be intensity associated with the affected image; the neural network model then compares the given image with the images present in the database.

Segmentation:

Given that, the brain tumor images and their corresponding masks are present in less number, 3929 samples each, we have used the Keras ImageData- Generator class in order to increase the dataset size. This process is also known as Data Augmentation. Also, in this step, we have resized the images into 256x256x3. The dataset is normalised before feeding it to the model.

Later, this data is fed into the deep neural network model for building a segmentation model.

3.2 Advantages

Deep Learning calculations specifically have vowed to change the establishment for dynamic and work processes, as these kinds of calculations can "learn" as a visual cue to execute an errand just as deciphering new information. In this manner, it could be conceivable by sending AI in radiology work processes to help wellbeing conveyance associations to acknowledge key operational and clinical results, for example

1. Assisting with improving the efficiency of clinical work processes using imaging.
2. Helping with bringing down the danger of "negative" clinical results related to delays in radiologist perusing, deciphering, and detailing
3. Enabling consideration groups to effectively see radiology work items, quicken clinical dynamic, and smooth out work processes, assisting with bringing about an improved patient encounter and results.

3.3 System Requirements

It is always evident that building deep learning algorithms requires good computational capabilities. Adhering to that, we have used the Windows system with 16GB of memory, which is acquainted with Intel i7 9th generation processor, more importantly 6 gigabytes of NVIDIA GTX 1660 Ti. Although, building the models with less number of layers, specifically less number of FLOPS (Floating Point Operations Per Second) which are the main reason for the necessity of high computation, needs a decent set of hardware requirements in order to have a smooth working of the project.

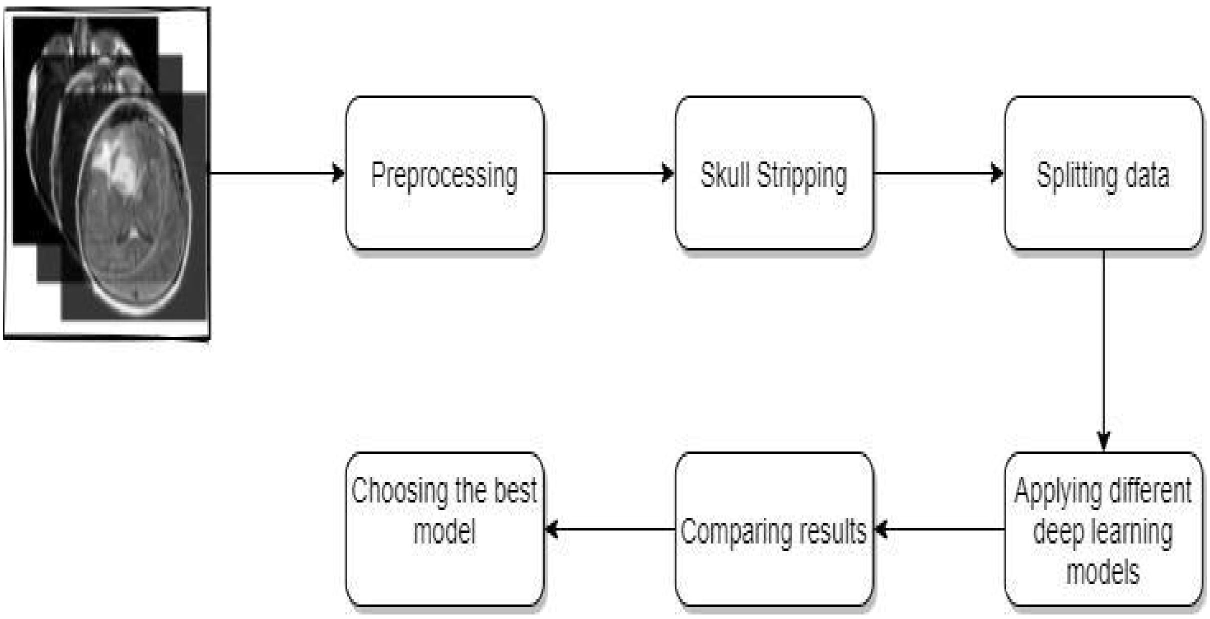
We have used python version 3.7, Anaconda distribution, Tensorflow Version 2.3.1, pandas, numpy, matplotlib, Flask. It is always recommended to try out different versions of the softwares as the latest versions of the software would have better performance.

CHAPTER 4

4. SYSTEM DESIGN

4.1 System Architecture

The proposed system is to develop an automatic system that has a built-in classification model and a segmentation model. The first part of building the model is pre-processing and then split into training, testing, and validation data. The data is further trained using multiple models and their respective results are compared after which the best model is used for further classifications. Similar procedure is carried out for building the segmentation model.



`

Figure 4.1 System Architecture-1

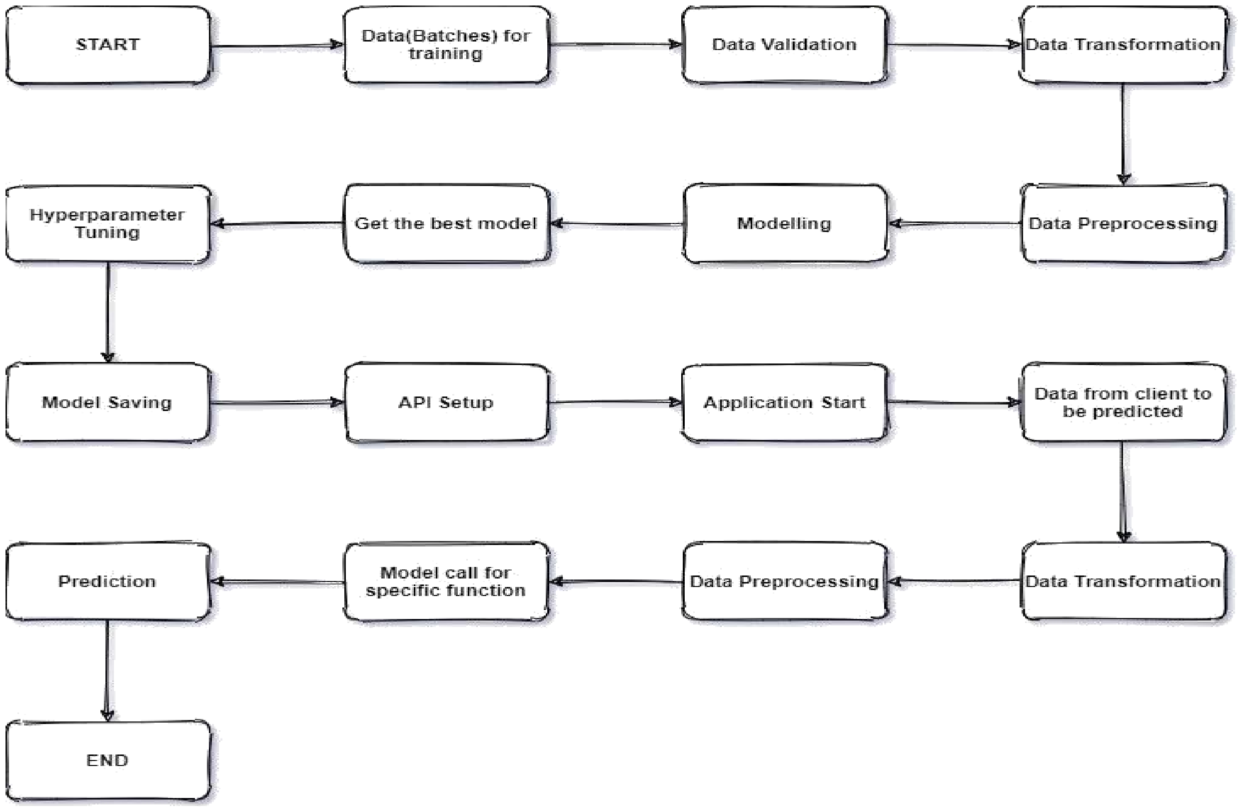


Figure 4.2 System Architecture-2

The proposed methodology[Figure 4.2] includes following tasks, in essence, dataset splitting, dataset transfor- mation, hyperparameter tuning, modeling. Further details of each task are discussed below.

4.2 System Modules

1) **Data Splitting:**

In general, there are various ways of splitting the data based on the given data. The dataset is spitted using hold-out technique, in essence 85% of training data and 15% of testing or validation data, considering the fact that the data set used in this study isn't biased and has almost equal amount of classes.Also, we have not used any stratification algorithms.

2) **Preprocessing of data:**

Digital pictures vary significantly in terms of picture size. Hence all the images are scaled to a common frame of reference depending upon the model we are using. As our emphasis is majorly on the brain region, unnecessary regions of the skull are removed (a.k.a., Skull

Stripping). In the end, the dataset is normalized using the Z-score. Once these steps are accomplished, the dataset is split into training, testing, and validation.

3) Building different Classification models:

Two deep learning models are built using VGG16 [20] and ResNet50. Although, the base model was built using the multilayer perceptron and used transfer learning in order to increase the performance of the model. The models are built by taking care of overfitting and underfitting cases while they are monitored and are evaluated using classification evaluation metrics. Early Stopping and loss monitoring call-backs were also used while training the model.

4) Segmentation Modeling:

In general, the Unet architecture proposed by Olaf Ronneberger is proven the best for segmentation tasks. The residual blocks in the unet architecture helps the CNN model to get significant results. In this study, we proposed a hybrid architecture blend- ing the residual blocks and the Unet architecture and we call it ResUnet.

4.3 UML Diagrams

Sequence diagram to understand the flow of the final outcome of the project.

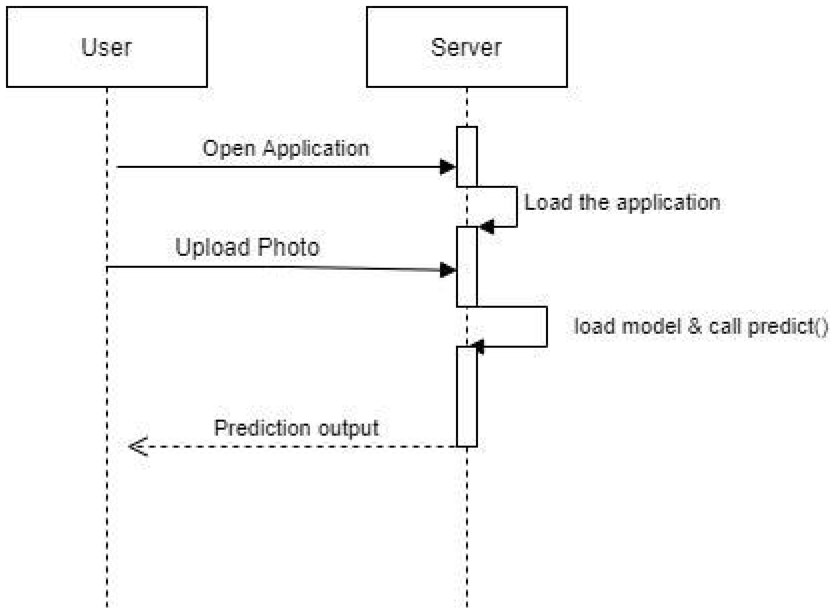


Figure 4.3 Sequence diagram

CHAPTER 5

5. IMPLEMENTATION

5.1 Algorithms

For the classification of the tumor our project is implemented on magnetic resonance imaging, medical images taken from lgg-segmentation dataset provided by TGCA. Our main goal is to process the dataset image using several techniques like resizing the image, classification is the process advocated for identification. The primary step of this project is to identify the tumor from the image extracted from the dataset. The basic idea must be preprocessing. In this preprocessing the color of the imaging is converted into grayscale image. It is obvious to extract the features from the grayscale image. The constituent values range from 0 to 255. And next, we have to link the properties of the other parts nearer from the center from the image as a whole. We need to adjust the values of the intensities of the converted image. All these efforts will raise the excellence of the complete image. Normally the value of the intensity of a cancer cell on the far is a traditional vegetative cell.

Convolutional Layer

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning calculation that can take in an information picture, assign significance (learnable loads and predispositions) to different viewpoints/objects in the picture, and have the option to separate one from the other. The pre-handling needed in a ConvNet is a lot lower when contrasted with other grouping calculations. While in crude techniques channels are hand-designed, with enough preparation, ConvNets can get familiar with these channels/qualities.

Activation layer:

The feature maps from a convolutional layer are fed through nonlinear activation functions. This makes it possible for the entire neural network to approximate almost any nonlinear function. The activation functions are generally the very simple rectified linear units, or ReLUs, defined as ReLU(z) = max(0, z), or variants like leaky ReLUs or parametric ReLUs. Feeding the feature maps through an activation function produces new tensors, typically also called *feature maps*.

Pooling

A pooling layer is another building block of a CNN.Its function is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network. Pooling layer operates on each feature map independently.The most common approach used in pooling is max pooling.

A 2x2 matrix is placed on a feature map in Max pooling and picks the highest value in that box. By picking the largest values from each iteration, a new matrix will be formed by these values known as pooled feature maps. The max pooling is used because it helps retain the fundamental features of an image while minimizing its size. It helps to minimize overfitting.

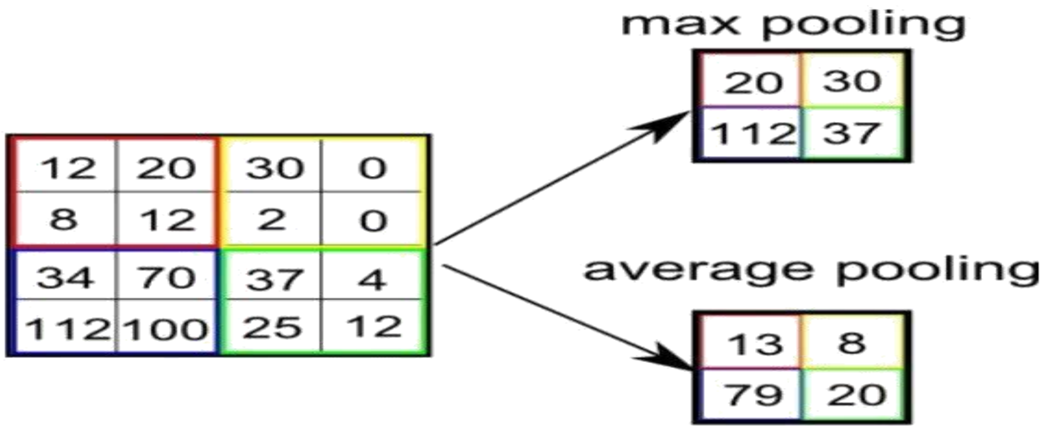


Figure 5.1: Max Pooling

Flattening Layer

Once we get the featured map from pooling, the next process is to flatten it. In this process we will be converting the entire pooled featured map into one single dimension, and this dimension will be for further processing by Neural Network.

Fully Connected Layer

In this process, the map which is generated from the flattening is set to a neural network. Different layers are present in this process mainly, the output layer. The fully connected layer in CNN is absolutely the same as the hidden layer in the ANN but in this it's connected to the fully connected layer. The predicted classes are obtained in the output layer. The network considers this information as input and error prediction is computed. These errors are thereafter passed back to the system for better accuracy. The function which is used to bring down the values is known as Softmax function.

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Figure 5.2 Softmax Function formula

VGG16

VGG-16 is a Convolutional Neural Network consisting of 16 layers(Refer Figure 5.3) in total which was proposed by K. Simonyan and A. Zisserman in 2014. The model follows the arrangement of Convolutional layers followed by Maxpooling layers consistently throughout the architecture. At the end it has two Fully connected layers followed by Softmax layer for output. VGG16 has a total of 138 million parameters. The important point to note here is that all the conv kernels are of size 3x3 and maxpool kernels are of size 2x2 with a stride of two.

Three Fully-Connected (FC) layers follow a heap of convolutional layers (which has an alternate profundity in various structures): the initial two have 4096 channels each, the third performs 1000-way ILSVRC arrangement and along these lines contains 1000 channels (one for each class). The last layer is the delicate max layer. The design of the completely associated layers is the equivalent in all organizations.

All concealed layers are outfitted with the correction (ReLU) non-linearity. It is likewise noticed that none of the organizations (aside from one) contain Local Response Normalization (LRN), such standardization doesn't improve the exhibition on the ILSVRC dataset, yet prompts expanded memory utilization and calculation time.

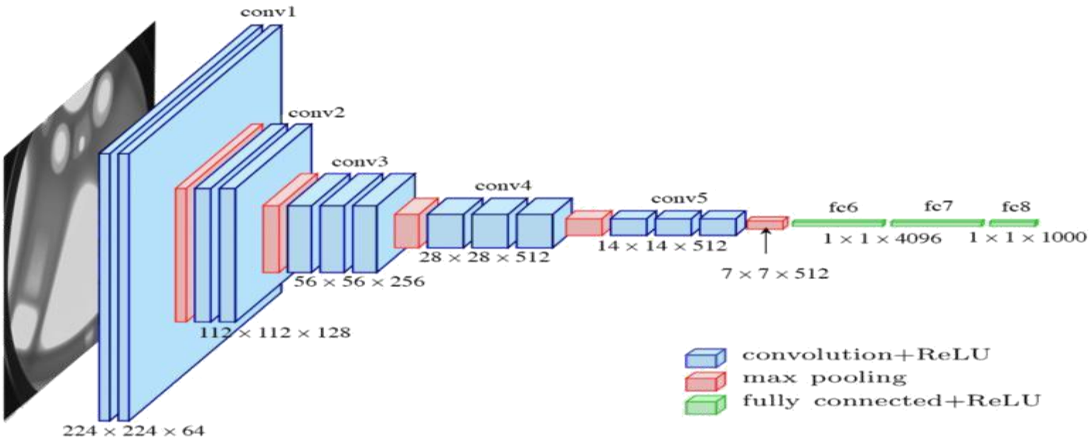


Figure 5.3: VGG16 Architecture

ResNet50

ResNet50 is a variant of the ResNet (Residual Network) model. It consists of 48 Convolutional layers along with one Max Pooling layer and one Average Pooling layer. It used relu (Rectified Linear Unit) as its activation function. The ResNet-50 model consists of 5 stages each with a convolution and Identity block. Each convolution block has 3 convolution layers and each identity block also has 3 convolution layers. The ResNet-50 has over 23 million trainable parameters.

Deep Residual Network is practically like the organizations which have convolution, pooling, initiation and completely associated layers stacked one over the other. The lone development to the straightforward organization to make it a lingering network is the personality association between the layers. The screen capture beneath shows the remaining square utilized in the organization. You can consider the to be association as the bended bolt beginning from the info and sinking to the furthest limit of the lingering block.

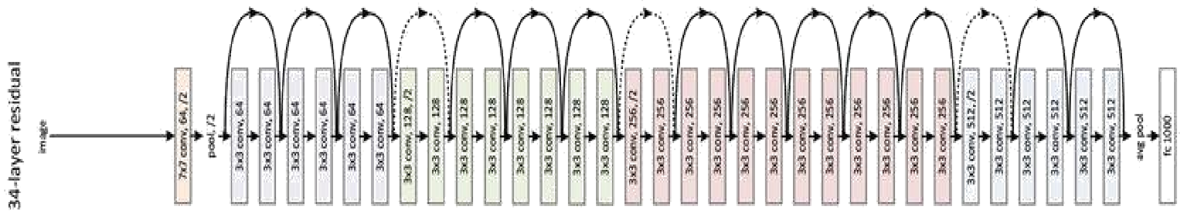


Figure 5.4 : ResNet Base architecture of 34 layers

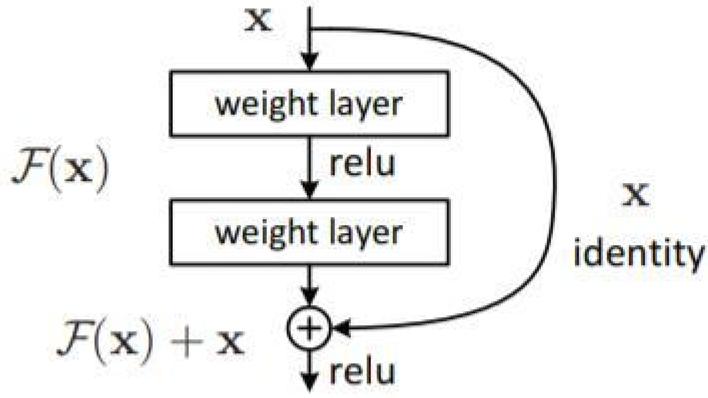


Figure 5.5 : Residual Learning Block

A residual Learning block is more or less a skip connection or shortcut connection , when the activation of a layer is skipped to a deeper layer in the neural network. Example of a residual block. As you can find in the figure above, the activation from a previous layer is being added to the activation of a deeper layer in the neural network.

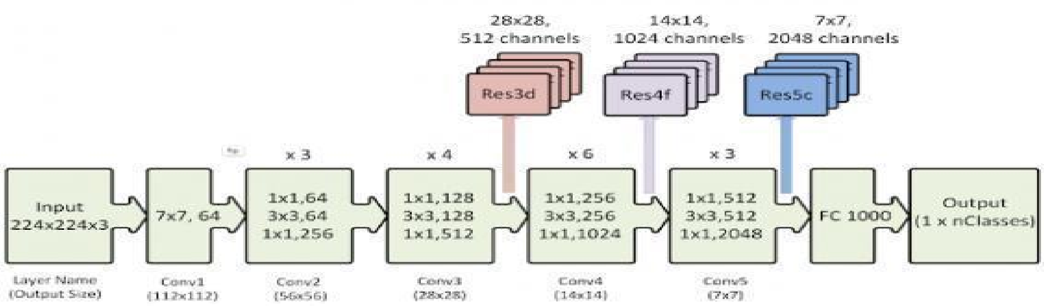


Figure 5.6 : ResNet50 Architecture

ResNext:

ResNext architecture(fig 2) combines UNet back- bone architecture with residual blocks to overcome the vanishing gradient descent problem present in the deep architectures. Unet is based on Fully Con- volutional Networks and modified in a way that it performs well on segmentation tasks. ResNnext consists of three parts

- Encoder or Contracting path
- Bottleneck
- Decoder or Expansion path

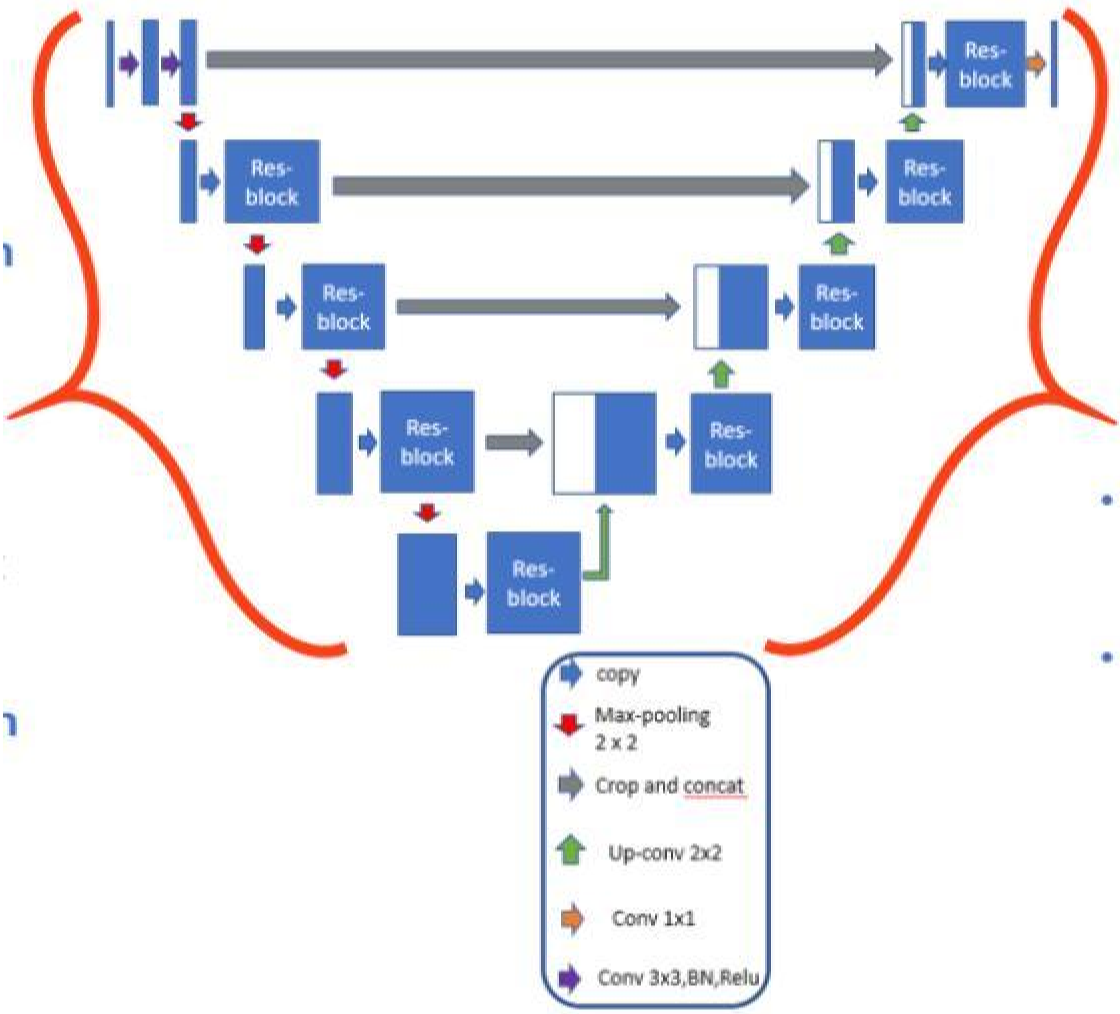


Figure 5.7 : ResNext Architecture

- **Encoder or Contracting path:**

The contraction path consists of several contraction blocks, each block takes an input that passes through residual blocks followed by several 2x2 MaxPooling layers. Feature maps after each block doubles, which helps the model learn complex features effectively.

- **Bottleneck:**

The bottleneck block serves as a connection between contraction path and the ex- pansion path.The block takes the input and then passes through a residual-block followed by 2x2 Up sampling convolution layers.

- **Decoder or Expansion path:**

Significant advantage of this architecture lies in the expansion or decoder path. Each block takes in the up-sampled input from the previous layer and concatenates with the corre- sponding output features from the residual blocks in the contraction path. This is then again passed through the residual block followed by 2x2 Up- sampling convolution layers.
This helps to ensure that features learning while contracting are used while contracting are used while reconstructing the image.Finally, in the last layer of the expansion path, the output from the residual block is passed through 1x1 conv layer to produce the desired output with the same as the input.

5.2 Implementation Steps

CNN is a multilayer network which is used to extract features from the data. It is mainly used in image processing. In this project we are going to discuss how a CNN model is used to detect and classify digital images having tumors. Once the CNN model is fully built, it can differentiate various objects of different images. All it requires is the input image to be given to the model. A collection of pixels form an image. Each pixel is determined by a value ranging from 0 to 255. Similar to ANN, CNN is inspired by the working of a human brain. CNN is mainly capable of classifying images by considering various features.

The initial phase of the implementation was data collection. Dataset of enlisted pictures together with manual division veils for each case utilized in consider is discharged and made freely available at the following link: https://kaggle.com/mateuszbuda/lgg-mri-segmentation

The second task was to build the dataset using the medical images that are available. The code used to do this is discussed in the source code section. After that, the preprocessing phase begins. Skull Stripping was the major phase of preprocessing. Now, in the modelling phase after understanding that the bottleneck features of the VGG16 architecture loses generalization when the architecture goes deeper, we had to use ResNet50 which performs better in terms of generalization as well as computation. Considering these two cases, we decided to use ResNet50 by adding a few bottleneck layers to it. Finally, the whole architecture comes down to 56 layers. The final output layer is used with Softmax activation function while all other layers use ReLU activation function.

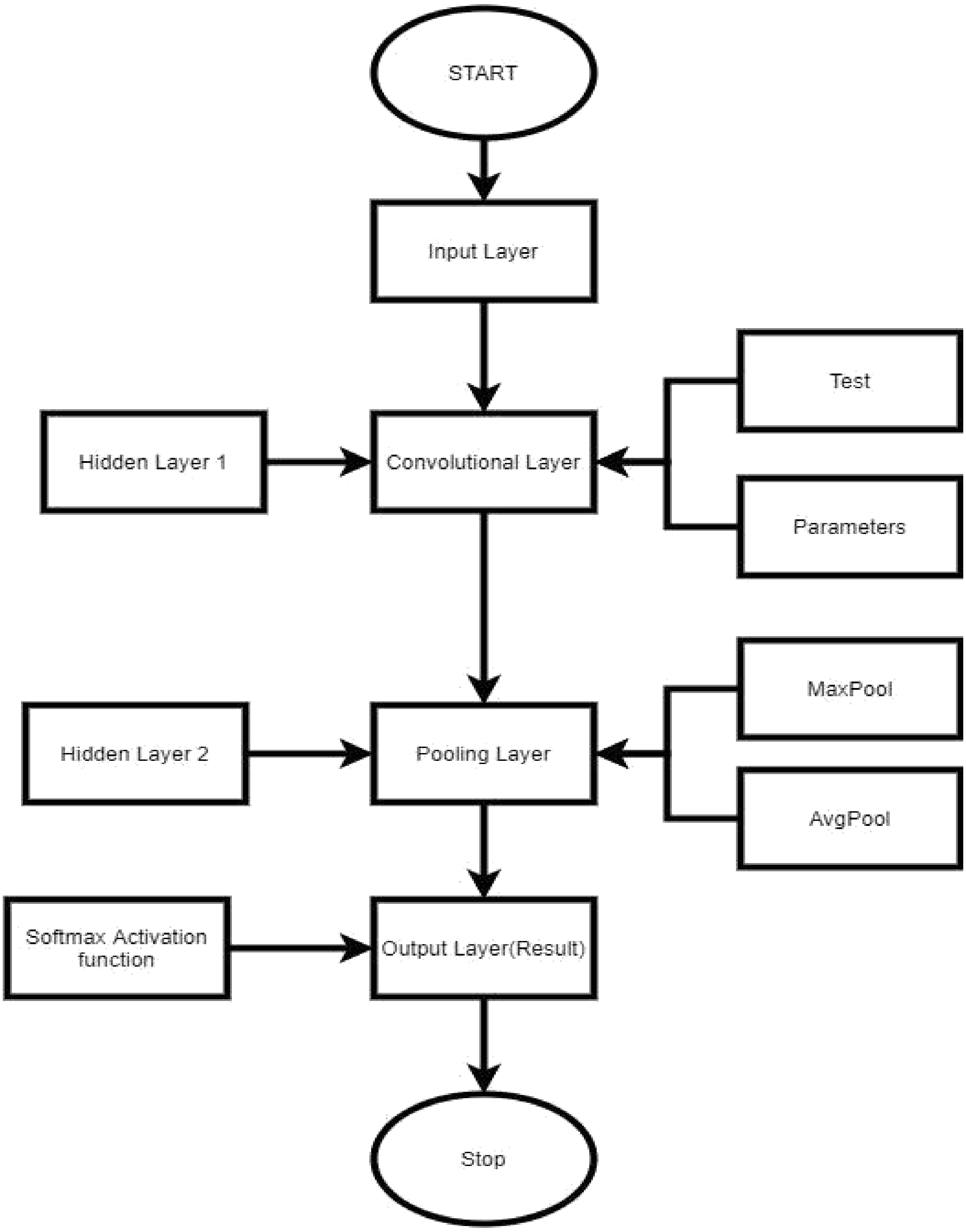


Figure 5.8 Flow Diagram

The above flow diagram represents the process of building the convolutional neural network model, where its layer is followed by a series of Max Pooling, Convolutional layers based on the architecture.

5.3 Source Code

Firstly, we need to convert the dataset which consists of images and their masks into the classification dataset by considering the pixel values. Below is the function, which does the described task.

```
def pos_neg_diagnosis(mask_path):
    value = np.max(cv2.imread(mask_path))
    if value > 0 :
        return 1
    else:
        return 0

brain_df['mask'] = brain_df['mask_path'].apply(lambda x: pos_neg_diagnosis(x))
```

Now, we have to split the data into train, test and validation using the keras DataGenerator class. Below is the sample code for the described process.

```
class DataGenerator(tf.keras.utils.Sequence):
  def __init__(self, ids , mask, image_dir = './', batch_size = 16, img_h =
256, img_w = 256, shuffle = True):

    self.ids = ids
    self.mask = mask
    self.image_dir = image_dir
    self.batch_size = batch_size
    self.img_h = img_h
    self.img_w = img_w
    self.shuffle = shuffle
    self.on_epoch_end()

  def __len__(self):
    'Get the number of batches per epoch'

    return int(np.floor(len(self.ids)) / self.batch_size)

  def __getitem__(self, index):
    'Generate a batch of data'

    #generate index of batch_size length
    indexes = self.indexes[index* self.batch_size : (index+1) *
self.batch_size]
```

```
    #get the ImageId corresponding to the indexes created above based on
batch size
    list_ids = [self.ids[i] for i in indexes]

    #get the MaskId corresponding to the indexes created above based on
batch size
    list_mask = [self.mask[i] for i in indexes]


    #generate data for the X(features) and y(label)
    X, y = self.__data_generation(list_ids, list_mask)

    #returning the data
    return X, y

  def on_epoch_end(self):
    'Used for updating the indices after each epoch, once at the
beginning as well as at the end of each epoch'

    #getting the array of indices based on the input dataframe
    self.indexes = np.arange(len(self.ids))

    #if shuffle is true, shuffle the indices
    if self.shuffle:
      np.random.shuffle(self.indexes)

  def __data_generation(self, list_ids, list_mask):
    'generate the data corresponding the indexes in a given batch of images'

    # create empty arrays of shape (batch_size,height,width,depth) #Depth is
    3 for input and depth is taken as 1 for output becasue mask
consist only of 1 channel.
    X = np.empty((self.batch_size, self.img_h, self.img_w, 3))
    y = np.empty((self.batch_size, self.img_h, self.img_w, 1))

    #iterate through the dataframe rows, whose size is equal to the
batch_size
    for i in range(len(list_ids)):
      #path of the image
      img_path = str(list_ids[i])

      #mask path
      mask_path = str(list_mask[i])

      #reading the original image and the corresponding mask image
      img = io.imread(img_path)
      mask = io.imread(mask_path)

      #resizing and coverting them to array of type float64
      img = cv2.resize(img,(self.img_h,self.img_w)) img =
      np.array(img, dtype = np.float64)
```

```
        mask = cv2.resize(mask,(self.img_h,self.img_w))
        mask = np.array(mask, dtype = np.float64)

        #standardising
        img -= img.mean()
        img /= img.std()

        mask -= mask.mean()
        mask /= mask.std()

        #Adding image to the empty array
        X[i,] = img

        #expanding the dimnesion of the image from (256,256) to (256,256,1)
        y[i,] = np.expand_dims(mask, axis = 2)

    #normalizing y
    y = (y > 0).astype(int)

    return X, y

train_data = DataGenerator(train_ids, train_mask)
val_data = DataGenerator(val_ids, val_mask)
```

Once the preprocessing is completed, we have to load the weights of the pretrained models in order to apply transfer learning.

```
def resblock(X, f):
    '''
    function for creating res block
    '''
    X_copy = X  #copy of input

    # main path
    X = Conv2D(f, kernel_size=(1,1), kernel_initializer='he_normal')(X)
    X = BatchNormalization()(X)
    X = Activation('relu')(X)

    X = Conv2D(f, kernel_size=(3,3), padding='same',
kernel_initializer='he_normal')(X)
    X = BatchNormalization()(X)

    # shortcut path
    X_copy = Conv2D(f, kernel_size=(1,1),
kernel_initializer='he_normal')(X_copy)
    X_copy = BatchNormalization()(X_copy)

    # Adding the output from main path and short path together
```

```
    X = Add()([X, X_copy])
    X = Activation('relu')(X)

    return X

def upsample_concat(x, skip):
    '''
    funtion for upsampling image
    '''
    X = UpSampling2D((2,2))(x)
    merge = Concatenate()([X, skip])

    return merge
```

Adding the final layers after removing the bottleneck layers

```
head = clf_model.output
head = AveragePooling2D(pool_size=(4,4))(head)
head = Flatten(name='Flatten')(head)
head = Dense(256, activation='relu')(head)
head = Dropout(0.3)(head)
head = Dense(256, activation='relu')(head)
head = Dropout(0.3)(head)
head = Dense(2, activation='softmax')(head)

model = Model(clf_model.input, head)
model.compile(loss = 'categorical_crossentropy',
              optimizer='adam',
              metrics= ["accuracy"]
```

Adding Callbacks to the model

```
# compling model and callbacks functions
adam = tf.keras.optimizers.Adam(lr = 0.05, epsilon = 0.1)
seg_model.compile(optimizer = adam,
                  loss = focal_tversky,
                  metrics = [tversky]
                 )
#callbacks
earlystopping = EarlyStopping(monitor='val_loss',
                              mode='min',
                              verbose=1,
                              patience=20
                             )
```

```
# save the best model with lower validation loss
checkpointer = ModelCheckpoint(filepath="ResUNet-segModel-weights.hdf5",
                               verbose=1,
                               save_best_only=True
                              )
reduce_lr = ReduceLROnPlateau(monitor='val_loss',
                              mode='min',
                              verbose=1,
                              patience=10,
                              min_delta=0.0001,
                              factor=0.2
                             )
```

Finally, training the model

```
h = seg_model.fit(train_data,
                  epochs = 60,
                  validation_data = val_data,
                  callbacks = [checkpointer, earlystopping,
reduce_lr]
                 )
```

CHAPTER 6

6. TESTING

We have performed two types of testing for the successful completion of the project. They are Integration testing and Unit Testing.

Integration Testing

After every unit is altogether tested, it is coordinated with different units to make modules or parts that are intended to perform explicit errands or exercises. These are then tried as gathering through joining testing to guarantee entire portions of an application carry on true to form (i.e, the cooperations between units are consistent). These tests are regularly outlined by client situations, for example, signing into an application or opening documents. Coordinated tests can be directed by either designers or free analyzers and typically include a mix of mechanized practical and manual tests.

Unit Testing

Unit testing is the principal level of testing and is regularly performed by the actual engineers. It is the way toward guaranteeing singular parts of a piece of programming at the code level are utilitarian and work as they were intended to. Designers in a test-driven climate will commonly compose and run the tests preceding the product or highlight being ignored to the test group. Unit testing can be led physically, yet mechanizing the cycle will accelerate conveyance cycles and grow test inclusion. Unit testing will likewise make investigating simpler in light of the fact that discovering issues prior methods set aside less effort to fix than if they were found later in the testing cycle.

CHAPTER 7

7. RESULTS

There are various evaluation metrics for the classification model. Here are the few that we have considered. In this study we have used Tversky loss function[16] in order to make evaluation more fault free. This loss function has parameters that can be optimized to get significant results by penalising the loss function.

This loss function contains constants ’alpha’ and ’beta’ to act as penalising factors in case of false positives and false negatives.In our study we have used

$\alpha = 0.7$

Accuracy:

Accuracy refers to the ratio of the instances that are correctly classified to the total number of instances. It is used to evaluate the classification model when the data is balanced. Here’s the formula

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

FN – False Negative, FP – False Positive, TN – True Negative, TP – True Positive.

2) **Recall:**

Recall refers to the ratio of true positives to the sum of true positives and false negatives. This evaluation metric is used as we want to check the efficiency of false negatives. Here’s the formula

Recall = TP / (TP + FN)

Confusion Matrix :

A Confusion matrix is a 2 X 2 matrix which quantifies about all the possible outcomes of the classification model, in essence, the true positive, true negative, false positive, and false negative. Refer below figure7.5

	precision	recall	f1-score	support
0	0.94	0.98	0.96	388
1	0.97	0.88	0.92	202
accuracy			0.95	590
macro avg	0.95	0.93	0.94	590
weighted avg	0.95	0.95	0.95	590

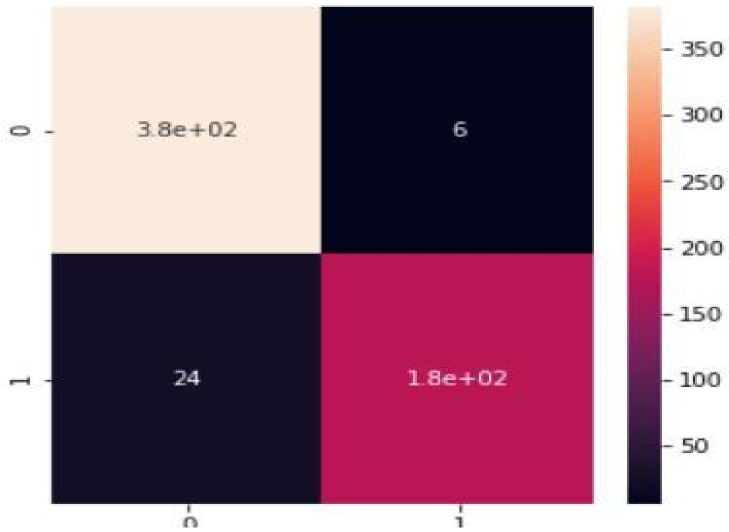


Figure 7.1 Classification Model Confusion Matrix

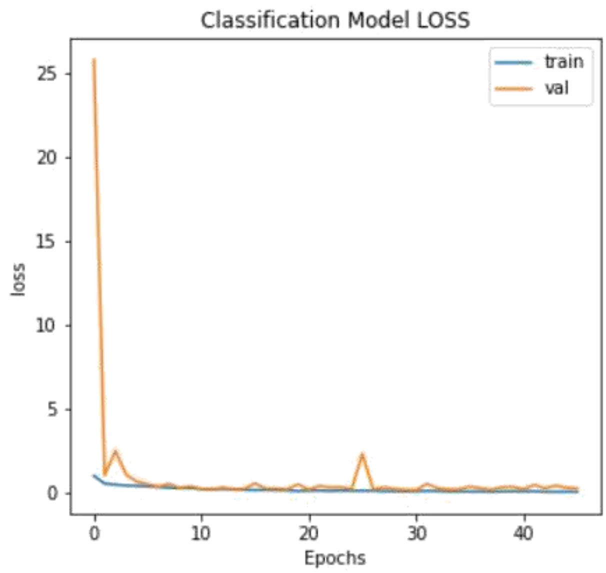


Figure 7.2 Classification Model Loss

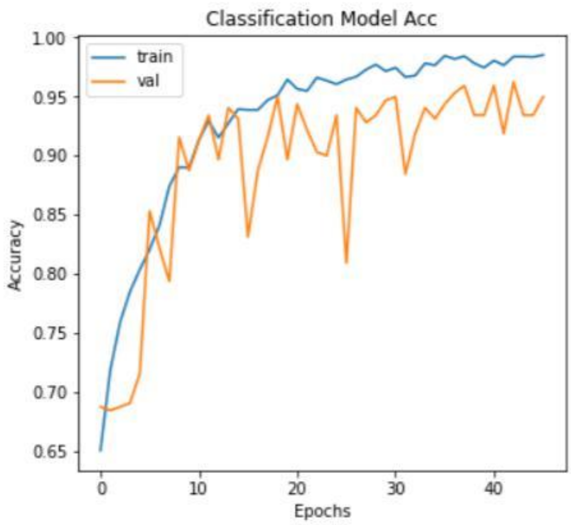


Figure 7.3 Classification Model Accuracy

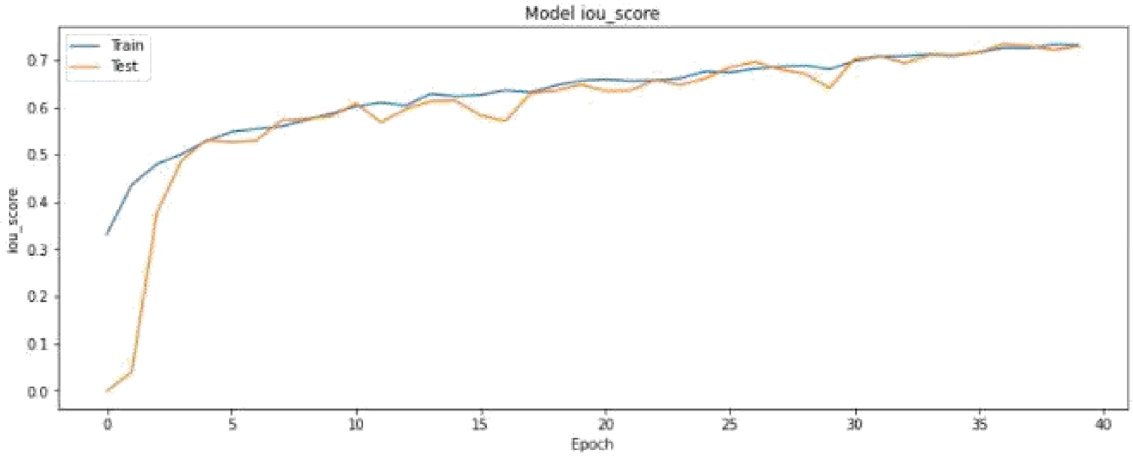


Figure 7.4 Segmentation Model Score(IOU)

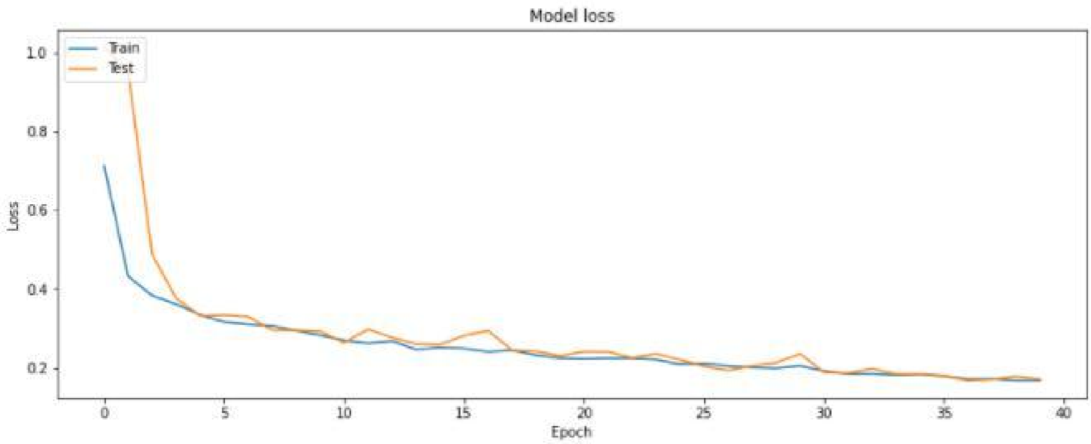
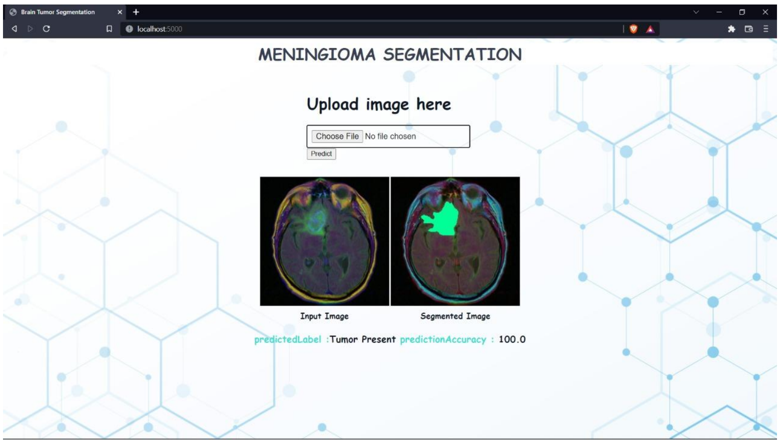


Figure 7.5 Segmentation Model loss(IOU loss)

The above results show that the obtained accuracy of the classification model is 94%. Also, Figure 7.2 shows that the loss of both the training and testing are significantly small by the end of training which means that the model is able to generalize well.

Apart from the accuracy metric, Recall is also important for this use case. The model achieved a recall value of 0.88 (refer Figure 7.1)which is close to 1, the desired recall value for classification.

After keen observation of fig7.4 and fig7.5, we can clearly understand that the model is accurate, achieving a significant trade-off between over-fitting and under-fitting. Although, we can still improve the model, given more digital images and computation



CHAPTER 8

8. CONCLUSION

In this project, we proposed a novel ResUnet architecture to extract more features efficiently on brain tumor segmentation data. The model architec- ture can be improved by adding more frameworks with multiple residual blocks at both the contraction and the expansion paths. Therefore, our future work is to look into more detailed features and also identify the grade of tumor based on the segmented tumor shape.

9. FUTURE ENHANCEMENTS

This work's objective was to segment the presence of the tumor in the given MRI image and locate it. Future work includes building of models that can improve the relationship between the shape features in identifying the grade of gliomas.

10. REFERENCES

[1] Cancer Genome Atlas Research Network. Comprehensive, integrative genomic analysis of diffuse lower-grade gliomas. New England Journal of Medicine, 372(26):2481–2498, 2015.

[2] Chang-Ming Zhang and Daniel J Brat. Genomic profiling of lower-grade gliomas uncovers cohesive disease groups: implications for diagnosis and treatment. Chinese journal of cancer, 35(1):12, 2016.

[3] G.Susmitha Valli " An approach to detect Bone Tumor Using Segmentation Technique" IJPR; ISSN No:0975–2366; 2018

[4] Susmitha Valli Gogula, K.Shekar, P.Purushotham , Y Teja Sai Bhavani "Android based image processor for blind", International journal of innovative technology and exploring engineering, Vol :9(1), Nov/2019 PP: 1229-1232

[5] Buda M, Saha A, Mazurowski MA (2019) Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. Comput Biol Med 109: 218–225

[6] Susmitha Valli Gogula, Ch. Divaker, Ch.Satyanarayana, Yedla Phani Kumar, vadapalli santhoshi lavanya "Computational investigation of PkcB inhibitors for the treatment of diabetic retinopathy", Journal of Bioinformation, 2013,Vol 9(20).

[7] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

[8] Maciej A Mazurowski. Radiogenomics: what it is and why it is important. Journal of the American College of Radiology, 12(8):862–866, 2015..

[9] Ian Goodfellow. Deep learning of representations and its application to computer vision. 2015

[10] Anastasia Ioannidou, Elisavet Chatzilari, Spiros Nikolopoulos, and Ioannis Kompatsiaris. Deep learning advances in computer vision with 3d data: A survey. ACM Computing Surveys (CSUR), 50(2):20, 2017.

[11] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[12] Yann LeCun and Yoshua Bengio. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 3361(10):1995, 1995.