

Brain Image Segmentation using Machine Learning

Ministry/Organisation Name: Dept. of Atomic Energy

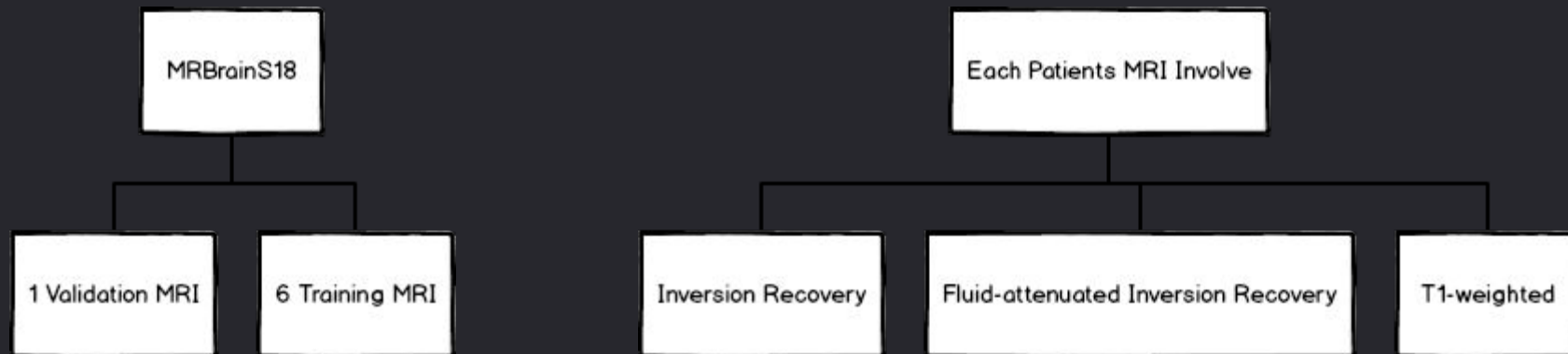
College Name: IIT (ISM) Dhanbad

Problem Statement: Brain Image Segmentation using Machine Learning

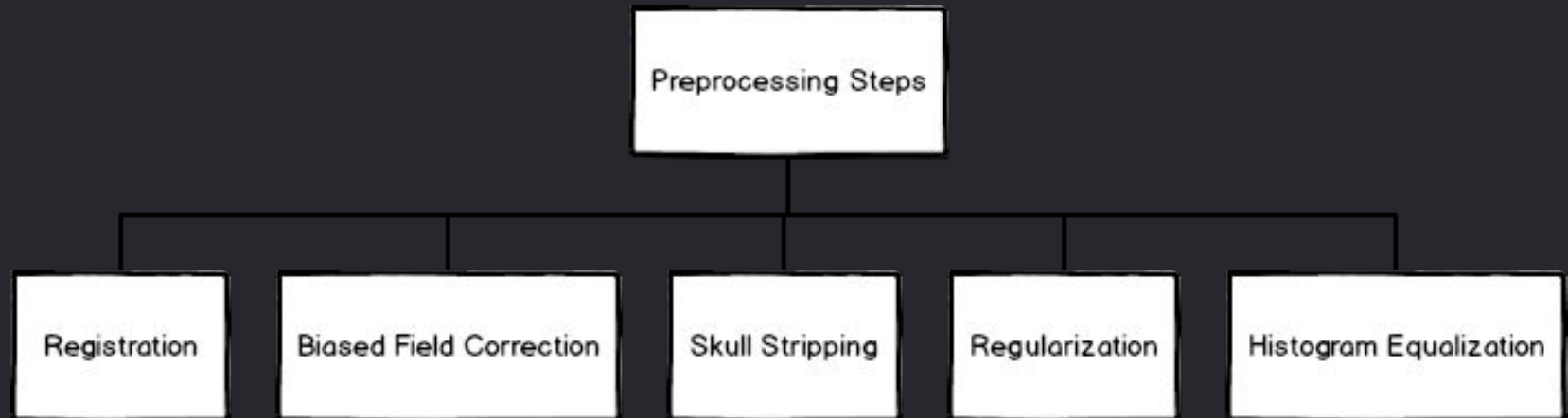
Team Name: Defaulting



Data Explanation



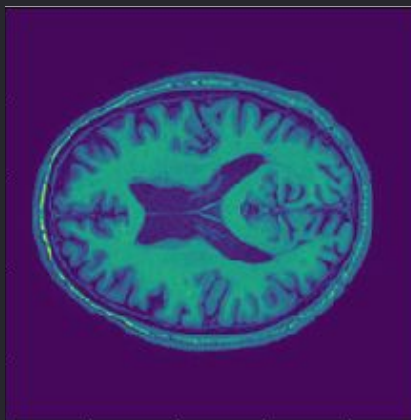
Preprocessing



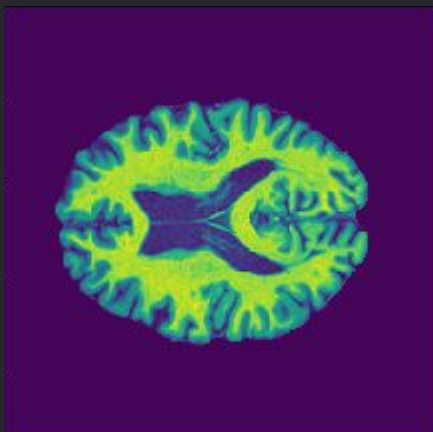
Preprocessing

- MRBrainS18 dataset came with Registered and Bias Field Corrected MRI's
- Skull Stripping:
We performed skull stripping only for T1 weighted MRI using DeepBrain library which creates a mask for skull removal with the help of skimage. The main reason for skipping skull stripping for IR and FLAIR was that they predominantly contain lesions which may be lost during skull stripping
- Histogram Equalization:
Performed on T1 weighted MRI to increase the contrast of Image

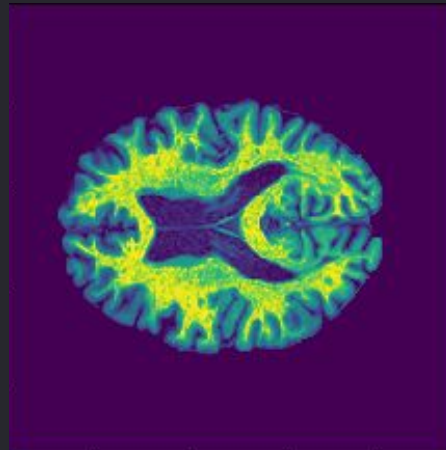
Preprocessing



Regularized Biased
Field Corrected MRI



Removed Skull

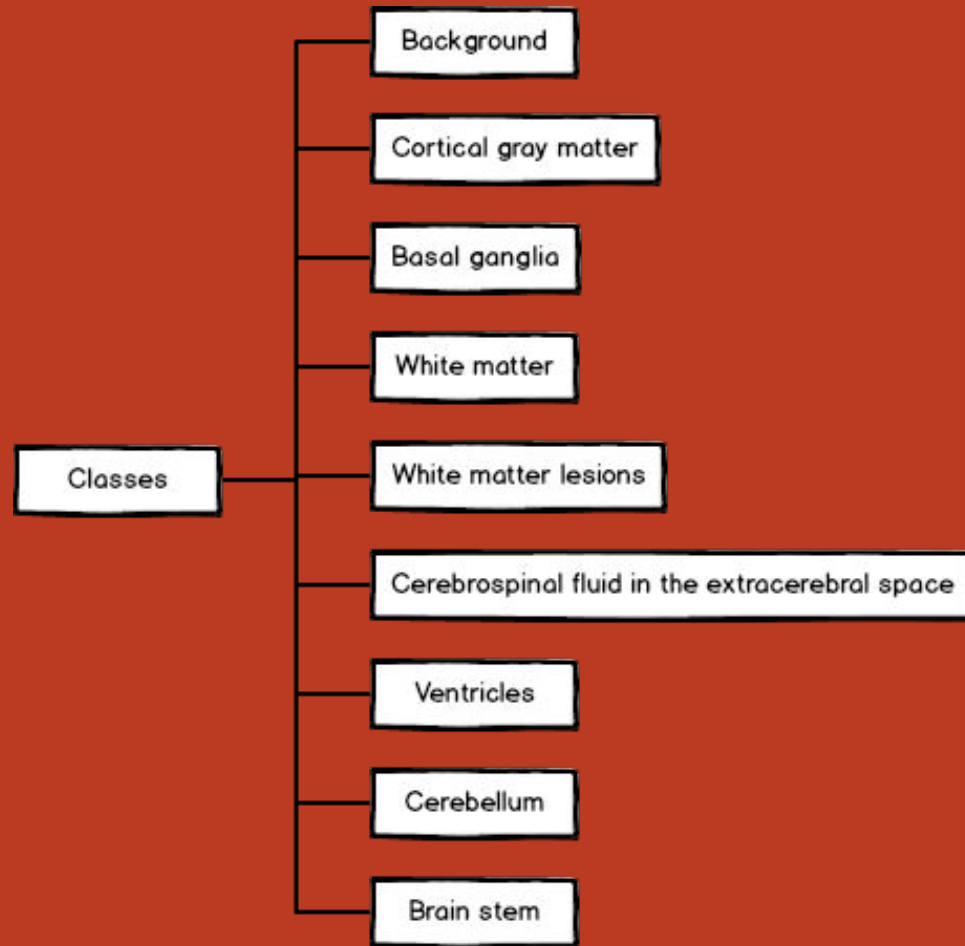


Histogram
equalization

Training



Methodology

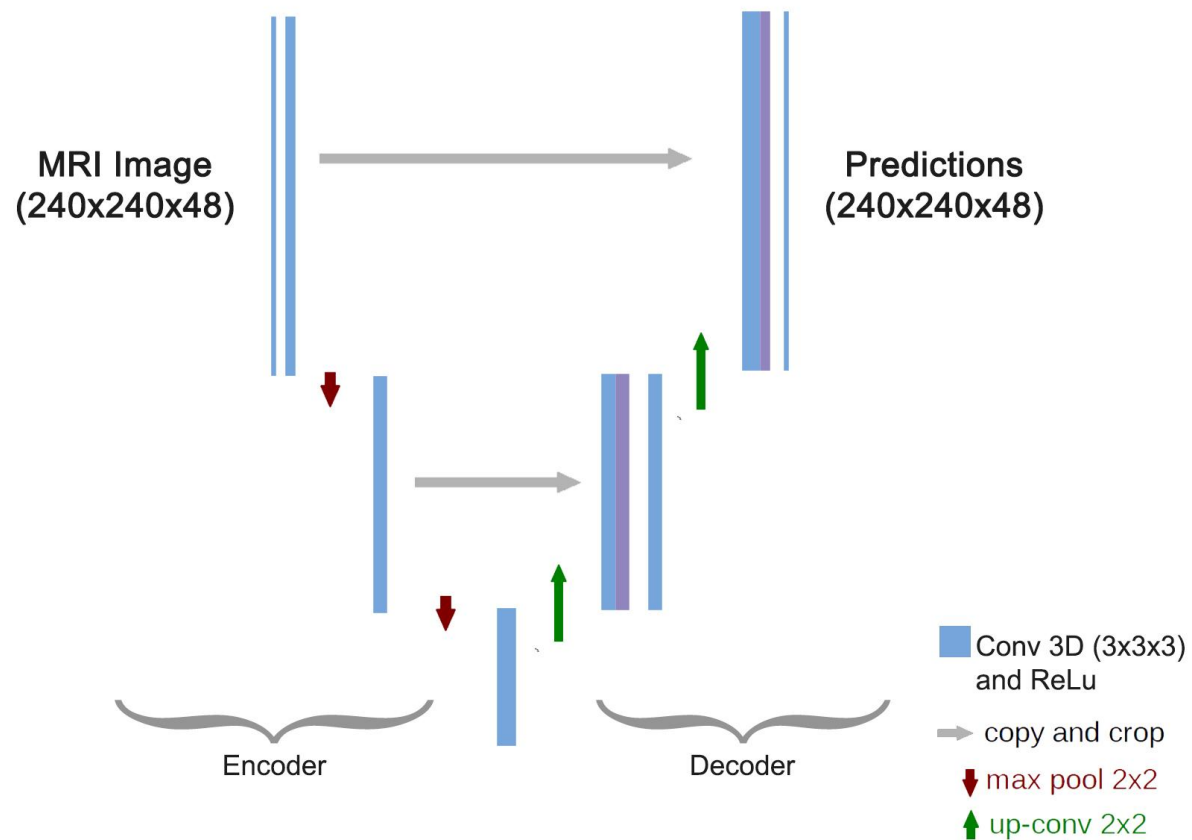


Methodology

We followed two different types of approaches for training

- Cortical gray matter, White matter, Cerebrospinal fluid in the extracerebral space can be easily reduced from T1- weighted MRI by applying threshold but the results were not good so we trained a small neural network to denoise the thresholding.

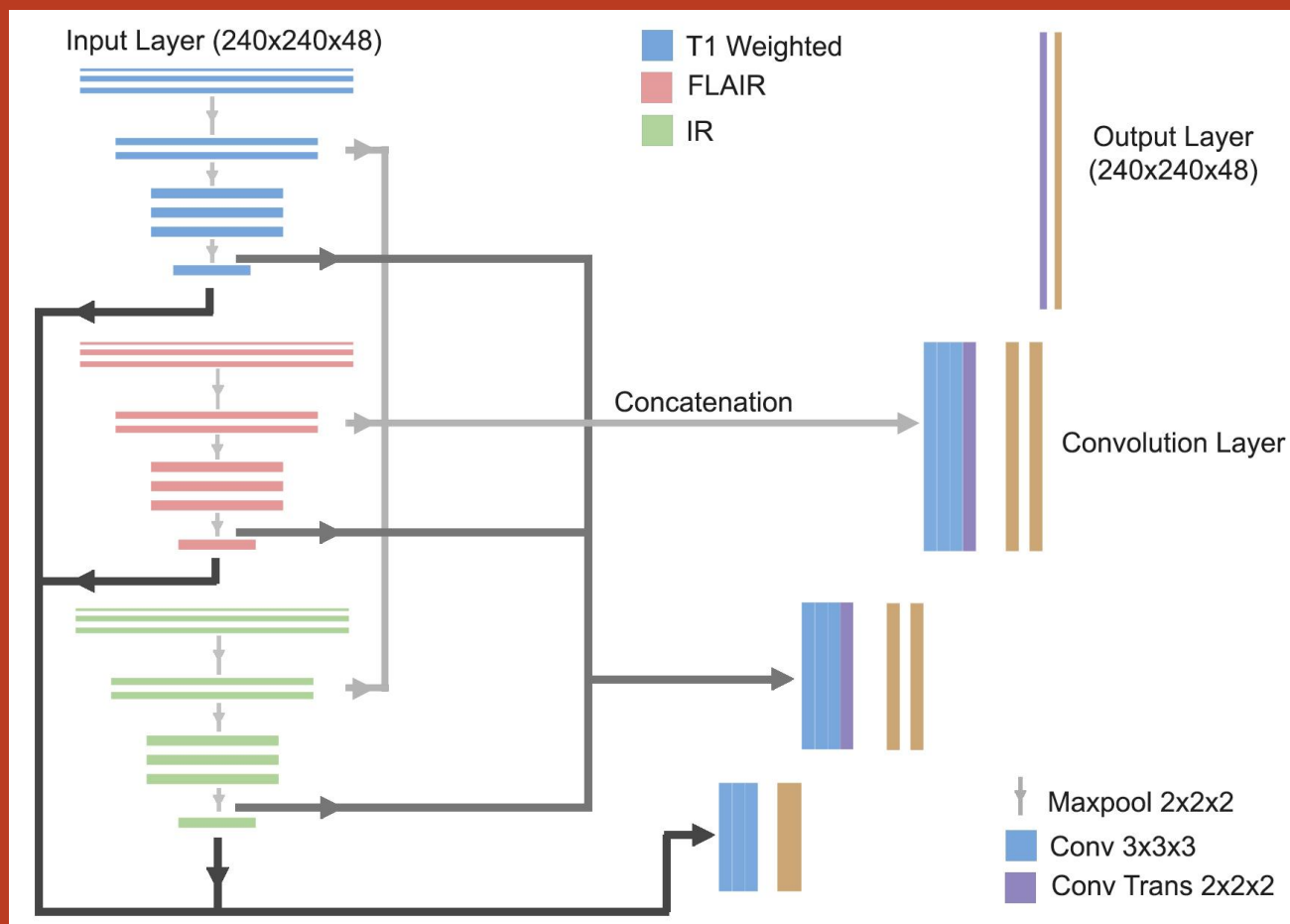
Network



Methodology cont...

- For rest of the classes we trained a custom model inspired by Unet Architecture. The model has 3 encoders stacked together in bottleneck layer and then a single decoder. There are skip connections from encoder to decoder to enhance segmentation.

Network



Difference between Unet and our Architecture

Unet Architecture

1. Only one encoder and one decoder
2. Deep architecture with about 10 Million parameters
3. Doesn't have dilated convolution layers

Our Architecture

1. Three encoder and one decoder
2. Shallow with about 600 Thousands parameters
3. Have dilated convolution layers

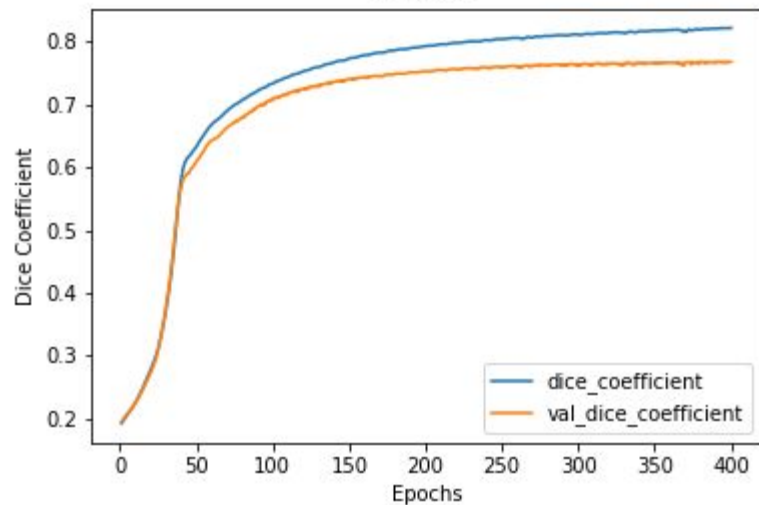
Loss Function

We took one minus dice coefficient as the loss function.

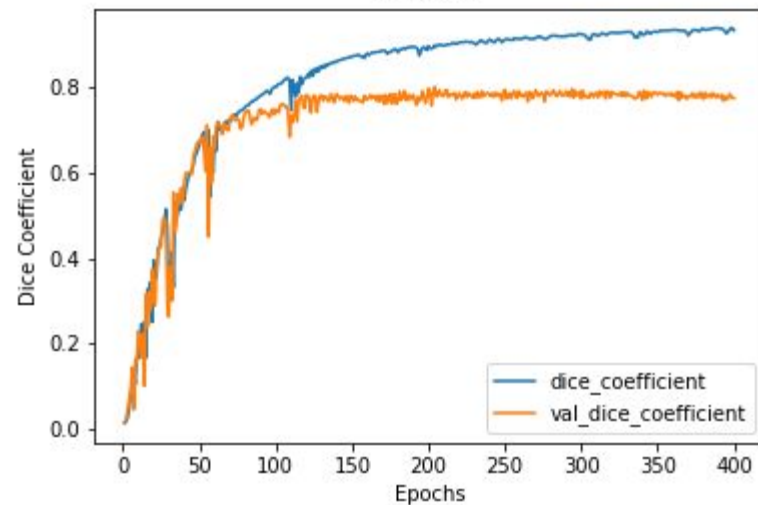
$$1 - \frac{2 * |X \cap Y|}{|X| + |Y|}$$

Training Curves

LABEL 1

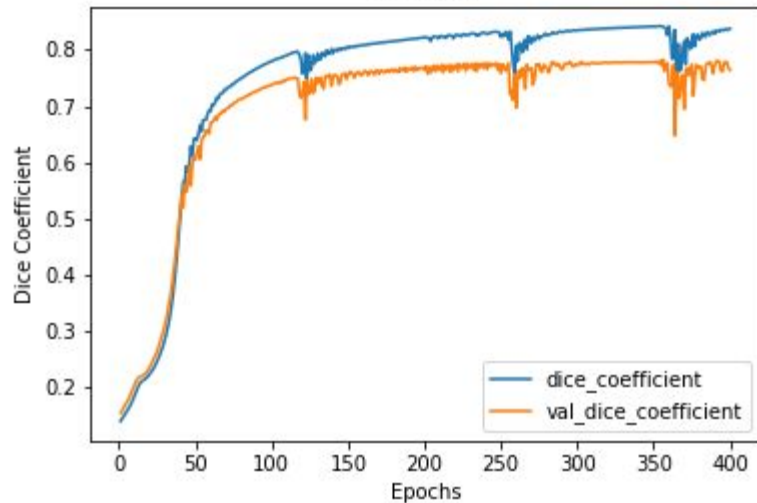


LABEL 2

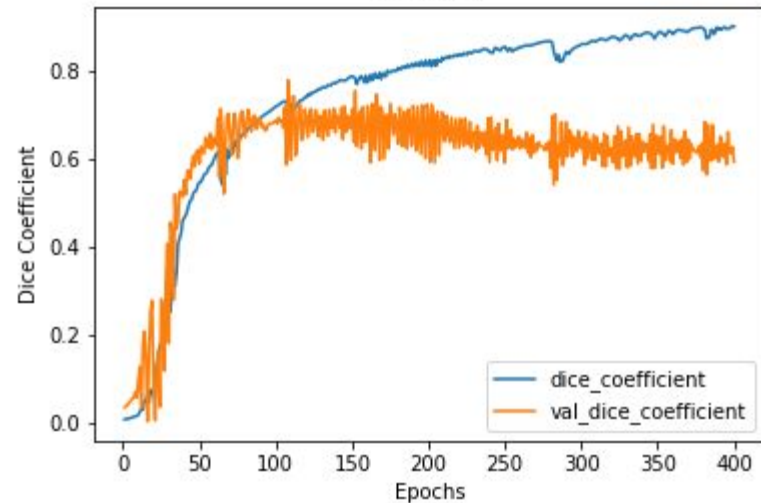


Training Curves

LABEL 3

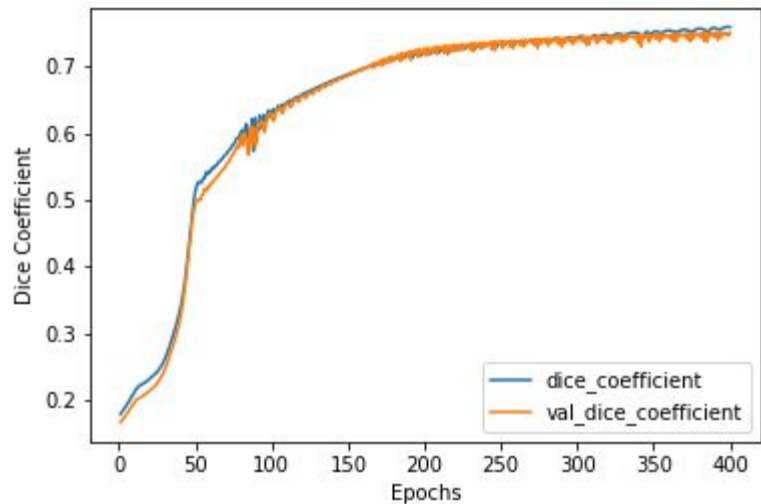


LABEL 4

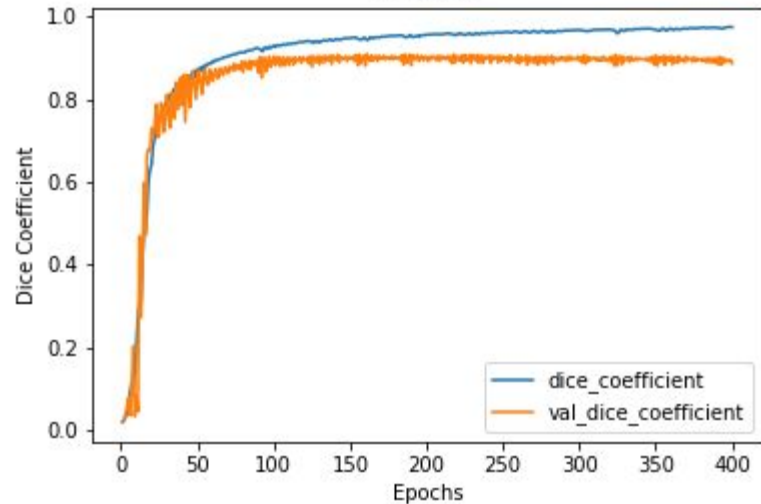


Training Curves

LABEL 5

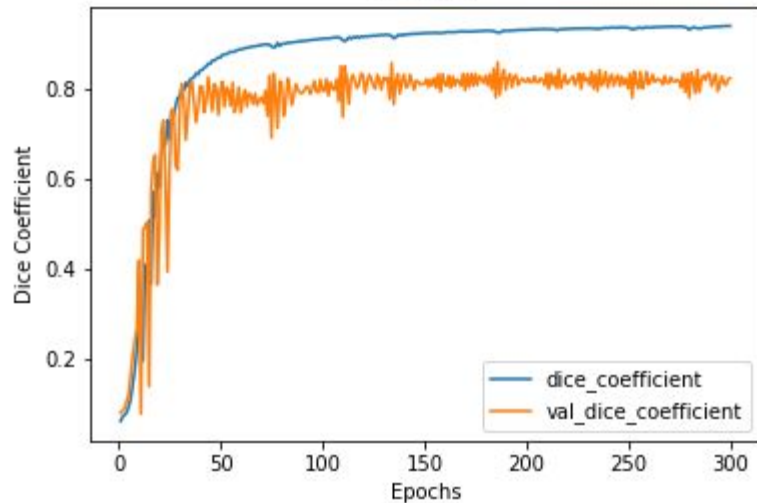


LABEL 6

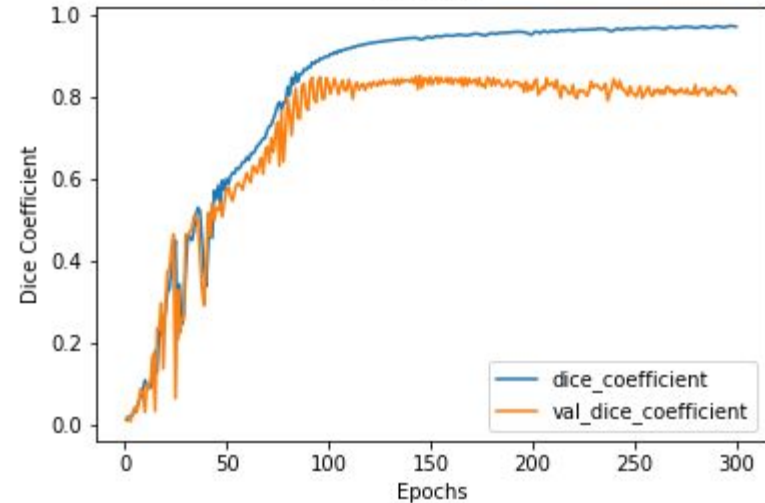


Training Curves

LABEL 7



LABEL 8



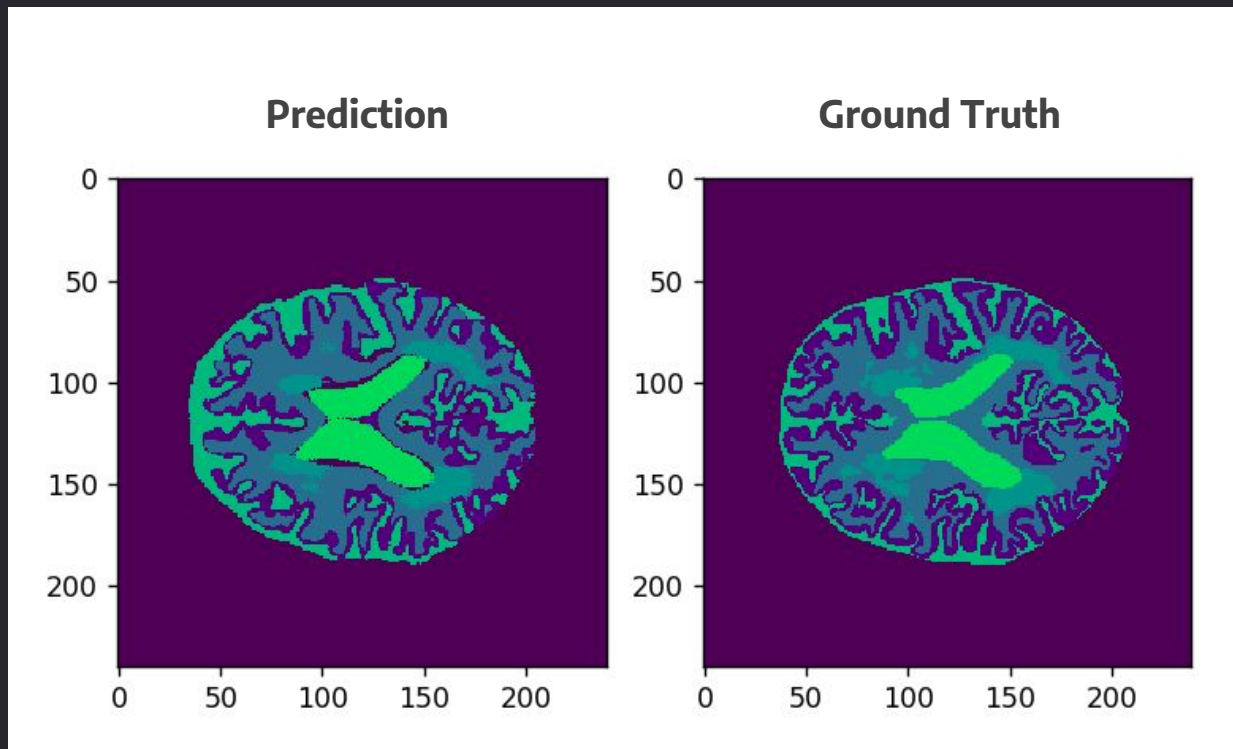
Findings

- Large network seems to overfit the data
- Skull removal improve dice's coefficient
- Thresholding works better for label 1,3 and 5

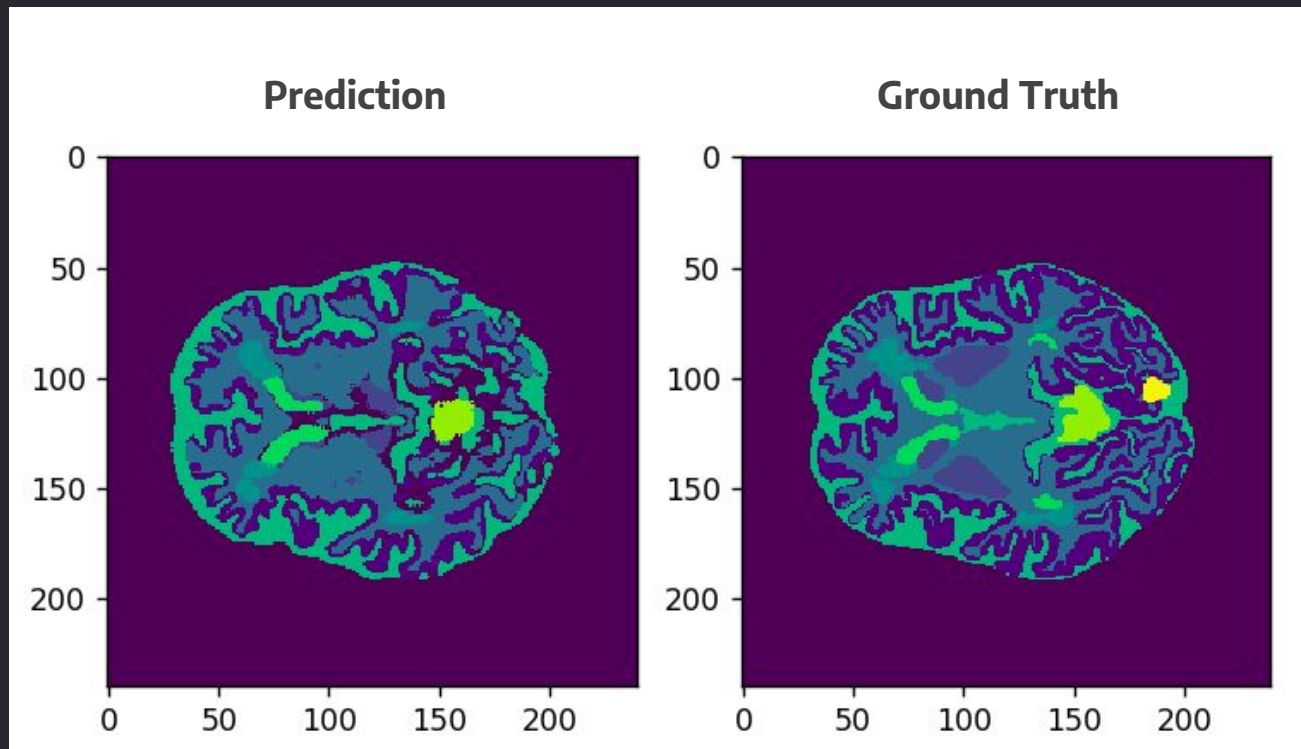
Problems

- Edge of tissue boundary is not clear
- Some small labels are difficult to segment
- There are many neighbour between different labels
- Lack of computation power

Results



Results



Evaluation

Class	Dice Score	Hausdorff Distance	Volume Similarity
1	0.7027	2.999	0.84702
2	0.7582	5.463	0.9835
3	0.7702	3.149	0.988
4	0.7463	9.054	0.8385
5	0.7043	3.149	0.986
6	0.8825	3.030	0.9955
7	0.8879	5.160	0.9207
8	0.8551	4.281	0.998

BraTS 2018 Dataset

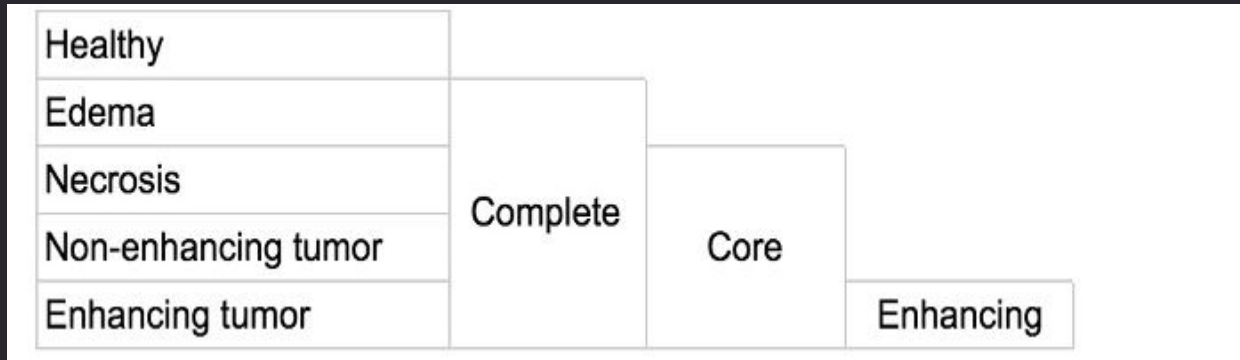
A more common dataset used in brain tumor segmentation is the BraTS Challenge 2018 dataset (Brain Tumor Segmentation Challenge).

Dataset size: 220 subjects with high grade tumors. 54 subjects with low grade tumors.

Input: For each subject we are given four MR images, i.e four three-dimensional volumes (FLAIR, T1W, T1C and T2).

Results

Labels: Each example has been segmented and labelled by 5 raters (certified doctors). Each voxel are labelled, (1) Healthy , (2) Edema , (3) Necrosis, (4) Non Enhancing tumour , (5) Enhancing tumour.



Approach

We used the same methods taken in the previous dataset .Images were regularized and histogram equalization was done to make the features appear more prominent.Finally skull stripping was done to the images to remove the skull borders from the images.

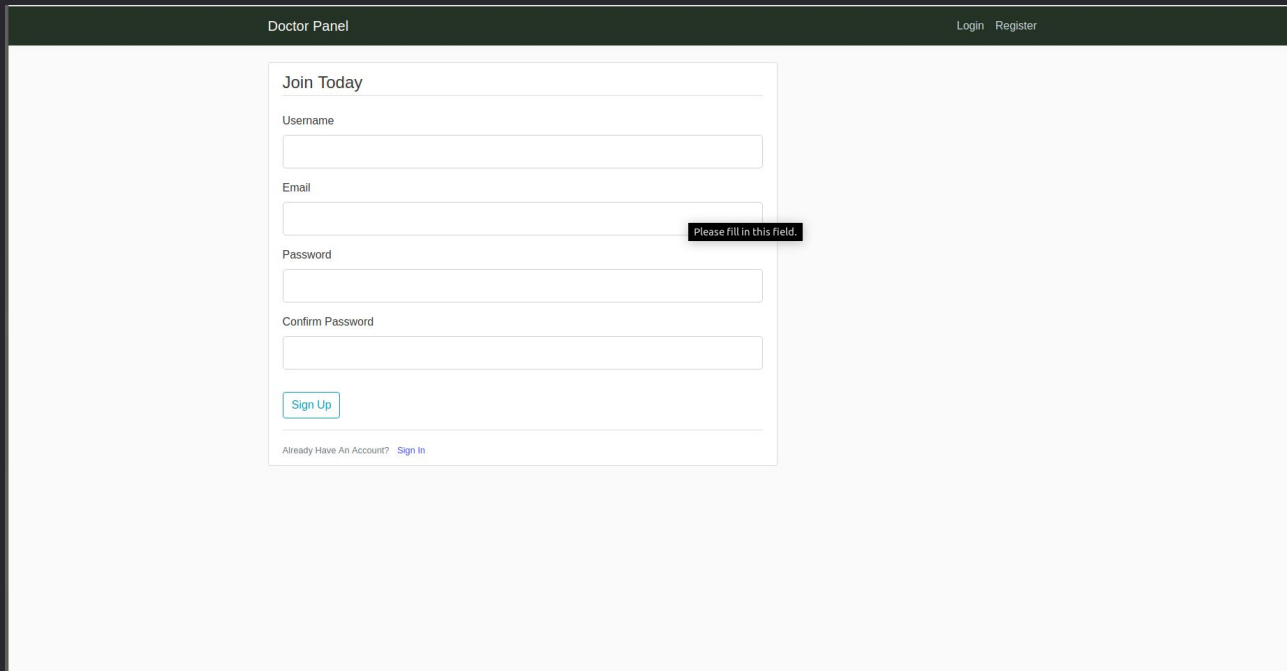
The same model consisting of a 3 Encoder U-Net Based architecture was used to train the data with the Loss function being **1 - Dice Coefficient**.

Challenges Faced

Since the dataset was a very large one , so the lack of time and computing resources made it very difficult for us to train the model in a short period of time.

To tackle this we instead of training on the whole dataset, we took the train and validation sample size of that equivalent to the MRBrainS18 dataset. This posed another problem to us that not all the samples that we took contained all the labels that we intended to predict

Registration page



The image shows a registration page with a dark green header. On the left of the header is the text "Doctor Panel" and on the right are the links "Login" and "Register". The main content area is white and contains a registration form titled "Join Today". The form has five input fields: "Username", "Email", "Password", and "Confirm Password". The "Email" field has a black tooltip with the text "Please fill in this field." pointing to it. Below the input fields is a blue "Sign Up" button. At the bottom of the form, there is a link "Already Have An Account? Sign In".

Doctor Panel Login Register

Join Today

Username

Email

Please fill in this field.

Password

Confirm Password

Sign Up

Already Have An Account? [Sign In](#)

Graphical User Interface

Fill patients Details

[Doctor Panel](#) [Add Patient](#) [View Patient](#) [Account](#) [Logout](#)

Patient Info

First Name

Gk

Last Name

Tejus

Age

19

Gender

☒ Male

☐ Female

Upload T1 File

Choose file

reg_T1.nii.gz

Upload FLAIR File

Choose file

FLAIR.nii.gz

Upload IR File

Choose file

reg_IR.nii.gz

Y_True File [Optional]

Choose file

segm.nii.gz

Add

Graphical User Interface

View existing patients details

The screenshot displays a web application interface for viewing patient details. At the top, a dark green navigation bar contains the text "Doctor Panel" and "Add Patient View Patient" on the left, and "Account Logout" on the right. The main content area features a light blue box titled "Patient Info". Inside this box, the patient's name "Gk Tejus" is displayed in bold. Below the name, the age "19 yrs" and gender "M" are listed. A section labeled "Download Previous uploaded file" contains four teal buttons: "Download T1 file", "Download FLAIR file", "Download IR file", and "Download y_True file". Below these buttons, a section labeled "Test on our Model" contains a purple button labeled "Make Prediction". At the bottom left of the page, a small text string "127.0.0.1:5000/analysis/TejusGk" is visible.

Doctor Panel Add Patient View Patient Account Logout

Patient Info

Patient Name:- **Gk Tejus**

Age:- 19 yrs

Gender:- M

Download Previous uploaded file

Download T1 file Download FLAIR file Download IR file Download y_True file

Test on our Model

Make Prediction

127.0.0.1:5000/analysis/TejusGk

Compare Predictions to Ground truths

[Doctor Panel](#) [Add Patient](#) [View Patient](#) [Account](#) [Logout](#)


Select frame number

Select particular Class


Cortical gray matter

Submit

Y_Pred



Y_True



Technology Stack

