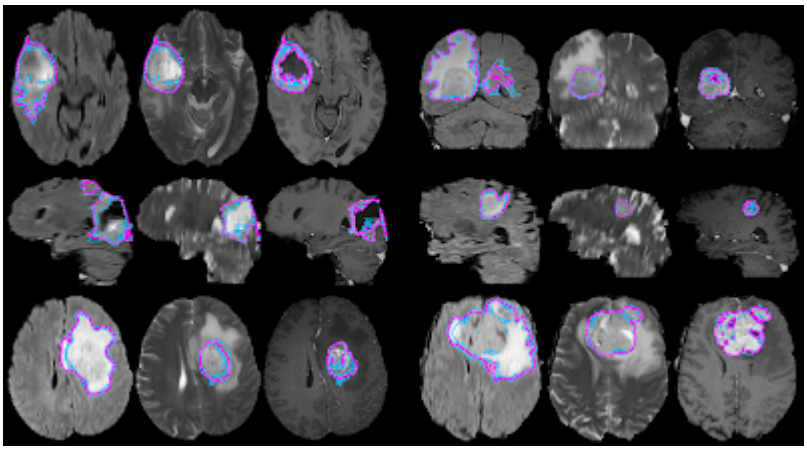
CAP5404 Deep Learning for Computer Graphics
Dr. Corey Toler-Franklin
Course Project Proposal - Comparing various pre-trained models for optimizing brain tumor detection using transfer learning.

Team Members:
Aadithya Kandeth - 69802791
Sai Nikhil Dondapati - 22286439
Shaanya Singh - 34762752

Abstract
Deep Learning for medical imaging is a fast growing field with widespread applications all across. Computer Aided Diagnosis is one of the main approaches which assists doctors in making correct decisions regarding their patients health. In this project we plan to approach the field of Neurology with the help of Deep Learning and find the most effective models that will give us the best possible accuracy on our dataset. We chose to use transfer learning as it is hard to find a large brain MRI dataset of which all are correctly annotated. Transfer Learning allows us to make our models more accurate even in the absence of larger datasets.

Objective
In this project, we aim to correctly classify brain MRI images as tumorous and non tumorous using various pre-trained models and transfer learning techniques.

Datasets
The datasets that we will be using for this project are the BraTS datasets introduced as a part of the Multimodal Brain Tumor Image Segmentation Challenge.



Source: https://sites.google.com/site/braintumorsegmentation/home/brats2015

Methodology
We plan to use architectures such as VGG-16, ResNet50, InceptionNet and InceptionNetV2 as the base models to be applied on the dataset. We then fine tune the pre-trained models of the

above mentioned architectures for the task at hand (Brain tumor detection). We will then compare the results of these techniques to find out which pretrained model performs best. We plan to use accuracy as a metric to compare the final results. The formula for accuracy can be given as:

$$\text{Accuracy} = \frac{\text{Number of correclty predicted images}}{\text{Total number of tested images}} \times 100\%$$

Expected Outcome
As we are using models that are trained on generic datasets and then fine tuning the weights and parameters for the specific use case of detecting brain tumors, we expect to come up with an architecture that generalizes very well for different brain MRI image datasets.

APIs to be used
Pytorch, Scikit-Learn, Matplotlib, Seaborn, OpenCV, Computer Vision Annotation tool

DL course topics
Machine Learning: The top layer of any complicated architecture we use for this task is a classification algorithm from Machine Learning domain.

Convolutional Neural Networks: Convolution Neural Networks tend to perform well for image classification. CNNs are an integral part of the architectures we plan to use for this project.

Recurrent Neural Networks: We plan to customize the various architectures mentioned above by adding an LSTM layer for identifying repetitive patterns in the image.

Preprocessing Techniques: Since the data size is limited, data augmentation has to be performed to increase the size of the training data.

Training/Testing and Fine tuning: To apply transfer learning, the model has to be fine tuned.

Timeline

Task	Estimted date of completion
Dataset Preparation	11/01/202
Implementing Baseline Models	11/10/2022
Finetuning pre-trained models	11/20/2022
Results & Conclusion	11/30/2022
Technical Report, Video Tour, Code package	12/07/2022

Work division

Preprocessing:
Data collection - Aadithya Kandeth
Data Augmentation - Sai Nikhil Dondapati
Cropping - Shaanya Singh

Architectures:
VGG-16 - Aadithya Kandeth
ResNet50 - Sai Nikhil Dondapati
InceptionNet - Shaanya Singh
InceptionResNetV2 - Aadithya Kandeth

Summary, Results, Technical writing:
Aadithya Kandeth
Sai Nikhil Dondapati
Shaanya Singh

References

- https://www.smir.ch/BRATS/Start2015
- https://arxiv.org/pdf/2102.01530.pdf
- https://arxiv.org/pdf/2106.07333.pdf
- https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a
- https://www.analyticsvidhya.com/blog/2021/10/understanding-transfer-learning-for-deep-learning/