Innovation Project Year 5 Report

MRI Stroke Segmentation

Prepared by HACHEM Mohamad
BOUGUERRA Aymen

Supervised by Prof. RODRIGUES Christophe

Project defended on January 26, 2023

ESILV YEAR 5 - 2022 / 2023

Introduction:

This project aims to utilize deep learning techniques for the segmentation of stroke lesions in magnetic resonance imaging (MRI) using a large, curated, open-source stroke neuroimaging dataset. The goal of us is to develop and evaluate an automated method for the accurate segmentation of stroke lesions in MRI images, which can aid in the diagnosis and treatment of stroke. We will employ deep learning techniques, such as convolutional neural networks (CNNs) and U-nets, to segment the stroke lesions in the MRI images.

State of the art:

The use of deep learning for the segmentation of stroke lesions in magnetic resonance imaging (MRI) has been an active area of research in recent years. State-of-the-art methods for segmenting stroke lesions using deep learning can be broadly grouped into two categories: fully convolutional networks (FCNs) and encoder-decoder architectures.

Fully convolutional networks (FCNs) were first introduced for image segmentation in 2015 by Long et al. in their paper "Fully Convolutional Networks for Semantic Segmentation" . FCNs are based on convolutional neural networks (CNNs) and are designed to take an input image and output a segmentation map. These networks have been used for the segmentation of stroke lesions in MRI images and have achieved good results.

Encoder-decoder architectures, such as U-Net, have been proposed in 2015 by Ronneberger et al. in "U-Net: Convolutional Networks for Biomedical Image Segmentation" . These architectures consist of an encoder network that extracts features from the input image and a decoder network that generates the segmentation map. These networks have been used for the segmentation of stroke lesions in MRI images and have shown to be effective.

In recent years, several variations and improvements of these architectures have been proposed, such as the use of attention mechanisms, and the use of pre-trained models for feature extraction. Some recent papers in 2020 and 2021, show that using 3D CNNs instead of 2D CNNs for the segmentation of stroke lesions in MRI images, have shown promising results.

Introduction to Strokes:

A stroke, also known as a cerebrovascular accident (CVA), is a serious medical condition that occurs when the blood supply to the brain is interrupted or reduced. This can happen in two ways: an ischemic stroke, which is caused by a blockage in a blood vessel supplying the brain, and a hemorrhagic stroke, which is caused by bleeding in the brain.

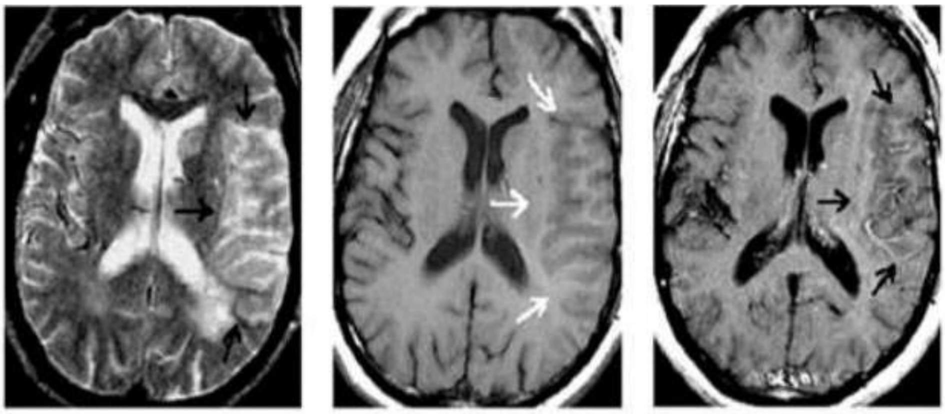
Symptoms of a stroke can vary depending on the type of stroke and the area of the brain affected. Common symptoms include sudden numbness or weakness on one side of the body, difficulty speaking or understanding speech, vision problems, and severe headache. Other symptoms can include confusion, difficulty walking, and loss of consciousness. If you suspect that you or someone you know is experiencing a stroke, it is important to seek medical attention immediately, as time is of the essence in treating a stroke.

Risk factors for stroke include hypertension, smoking, diabetes, high cholesterol, and a family history of stroke. Other risk factors include obesity, physical inactivity, excessive alcohol consumption, and certain medical conditions such as atrial fibrillation. To reduce your risk of stroke, it is important to manage these risk factors through lifestyle changes, such as eating a healthy diet, exercising regularly, and avoiding smoking.

Stroke can lead to long-term disabilities, including difficulty with movement, speech, and cognitive function. Rehabilitation and therapy can help stroke survivors regain function and improve their quality of life. It's important for stroke survivors and their loved ones to have realistic expectations about recovery and to be aware of the potential for long-term disabilities.

Introduction to MRIs

Magnetic Resonance Imaging (MRI) is a non-invasive medical imaging technique that uses a magnetic field and radio waves to produce detailed images of the body's internal structures. Unlike X-rays and computed tomography (CT) scans, MRI does not use ionizing radiation, making it a safer option for certain types of imaging. It is particularly useful for imaging soft tissue and organs, such as the brain, spine, joints, and internal organs, and is often used to diagnose and monitor a wide range of medical conditions.



Stroke MRI

The basic principle of MRI is the use of a powerful magnetic field to align the protons in the body's hydrogen atoms. When radiofrequency pulses are applied, the protons are stimulated and release energy, which is then detected by the MRI machine and used to create images of the body's internal structures. This process is called Nuclear Magnetic Resonance (NMR).

One of the major advantages of MRI is its ability to produce highly detailed images of soft tissue and organs, making it useful for a wide range of diagnostic and therapeutic applications. For example, it can be used to detect and diagnose tumors, cysts, and other abnormalities in the brain and spine. It is also useful for monitoring conditions such as multiple sclerosis and spinal cord injuries. Additionally, MRI is used to evaluate injuries and diseases of the joints, such as the knee, shoulder, and hip, as well as internal organs such as the liver, kidney, and prostate.

During an MRI, the patient is placed on a table that is inserted into the MRI machine, which is a large, tube-shaped device. The table then moves into the machine, and the patient is surrounded by the magnetic field. The patient will be asked to lie still during the procedure, which typically takes between 30 and 60 minutes. Depending on the type of scan, the patient may be given an injection of a contrast agent, which can help enhance the visibility of certain structures or blood vessels in the images.

MRI Machine

It's important to note that some patients may be contraindicated for MRI due to certain medical conditions or devices in the body, such as pacemaker or aneurysm clip. Additionally, patients who are pregnant or breastfeeding should not undergo MRI. It's always important to inform your doctor if you have any concerns or if you have any metal objects in your body before an MRI.

In conclusion, Magnetic Resonance Imaging (MRI) is a non-invasive medical imaging technique that uses a magnetic field and radio waves to produce detailed images of the body's internal structures. It is particularly useful for imaging soft tissue and organs, such as the brain, spine, joints, and internal organs, and is often used to diagnose and monitor a wide range of medical conditions. It's a safe and widely used imaging modality and is a valuable tool for medical diagnosis and treatment.

Types of MRIs

There are several different types of MRI, each with its own specific advantages and applications.

1. Traditional or Standard MRI: This type of MRI is the most commonly used and is the best at creating detailed images of soft tissue and organs such as the brain, spine, joints, and internal organs. Standard MRI uses a powerful magnetic field and radio waves to produce detailed cross-sectional images of the body.

2. Functional MRI (fMRI): This type of MRI can be used to detect changes in blood flow and brain activity. It allows doctors to see which areas of the brain are active during a specific task or function. It is used to study brain function, evaluate brain disorders, and identify areas of the brain that may be damaged or malfunctioning.

3. Diffusion MRI: This type of MRI can be used to study the movement of water molecules in the brain, which can be helpful in identifying changes in the white matter of the brain, such as those that occur in multiple sclerosis.

4. Magnetic Resonance Angiography (MRA): This type of MRI is used to create detailed images of blood vessels and can be used to detect blockages, aneurysms, and other blood vessel abnormalities.

5. Magnetic Resonance Spectroscopy (MRS): This type of MRI is used to study the chemical makeup of the brain and can be used to detect changes in brain chemistry that may occur in certain conditions, such as tumors and multiple sclerosis.

6. Magnetic Resonance Cholangiopancreatography (MRCP): This type of MRI is used to create detailed images of the biliary and pancreatic ducts, which can be used to detect abnormalities such as gallstones, tumors, and inflammation.
7. Cardiac MRI: This type of MRI can be used to create detailed images of the heart and can be used to detect heart disease, heart attack, and other cardiac conditions.
8. Breast MRI: This type of MRI is used to create detailed images of the breast and can be used to detect breast cancer and other breast abnormalities.

It's important to note that all types of MRI are safe and non-invasive, but patients who are pregnant or breastfeeding should not undergo MRI. Additionally, patients who are contraindicated due to certain medical conditions or devices in the body, such as pacemaker or aneurysm clip, should not undergo MRI. It's always important to inform your doctor if you have any concerns or if you have any metal objects in your body before an MRI.

Other types of scans:

There are several other types of medical imaging scans that can be used to diagnose and monitor a wide range of medical conditions, in addition to Magnetic Resonance Imaging (MRI). Some of these include:

X-ray: X-rays use ionizing radiation to produce images of the body's internal structures. They are often used to detect bone fractures, tumors, and other abnormalities in the bones and soft tissue.

Computed Tomography (CT) scan: CT scans use X-rays and computer technology to produce detailed cross-sectional images of the body. They can be used to detect tumors, blood clots, and other abnormalities in the body.

Ultrasound: Ultrasound uses high-frequency sound waves to produce images of the body's internal structures, such as the liver, kidney, and other internal organs. It can also be used to visualize blood flow and detect blood clots.

Positron Emission Tomography (PET) scan: This type of scan uses a small amount of radioactive material to produce images of the body's internal structures and detect changes in the body's metabolism.

Single-photon emission computed tomography (SPECT): This type of scan uses a small amount of radioactive material and a gamma camera to produce images of the body's internal structures and detect changes in the body's metabolism.

Digital Radiography (DR) or Digital X-ray: This type of scan uses X-rays and computer technology to produce detailed images of the body's internal structures. The images are captured digitally and can be displayed on a computer screen.

Fluoroscopy: This type of scan uses X-rays and a special imaging device to produce live, moving images of the body's internal structures. It can be used to diagnose and monitor a wide range of medical conditions.

It's important to note that each of these imaging modalities has its own unique advantages and disadvantages, and the best choice of imaging modality will depend on the specific condition being evaluated and the patient's individual needs.

Introduction to Deep Learning:

Deep learning is a subfield of machine learning that is inspired by the structure and function of the brain, specifically the neural networks that make up the brain. It involves training artificial neural networks, which are composed of layers of interconnected nodes or neurons on a large dataset. The networks learn to recognize patterns in the data and can be used for a variety of tasks such as image and speech recognition, natural language processing and decision making.

Deep learning algorithms are built using multiple layers of artificial neural networks, where each layer extracts a higher level of abstraction from the data. These layers are composed of interconnected neurons that are connected to each other and are used to process and transmit information. Each neuron receives input from the previous layer, performs a computation on that input, and then sends the result to the next layer. The process of training a deep learning model involves feeding it a large dataset and adjusting the weights of the connections between the neurons so that the model can accurately predict the output for a given input. This is done using a process called backpropagation, which involves computing the gradient of the error with respect to the model's parameters and updating the parameters to reduce the error.

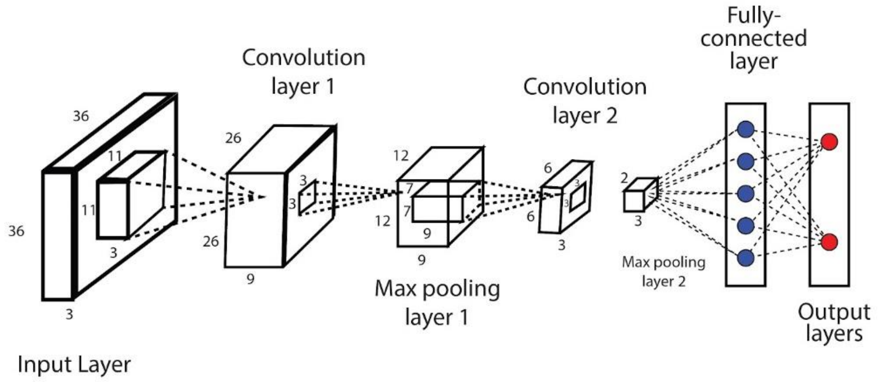
Deep learning models have been used to achieve state-of-the-art performance on a wide range of tasks, including image classification, object detection, and language translation. Deep learning models are particularly useful for tasks that involve unstructured data, such as images, videos, and audio. They can also be used to extract features from structured data, such as text, and use those features to improve the performance of other machine learning models. Deep learning algorithms require large amounts of data and computational resources to train. But recent advancements in hardware, such as graphics processing units, have made it possible to train large deep learning models in a reasonable amount of time. Additionally, pre-trained models and cloud-based platforms have made it easier for developers to use deep learning without needing access to large amounts of computational resources.

Introduction to CNNs:

A convolutional neural network (CNN) is a type of deep learning algorithm that is particularly well-suited for image recognition and processing tasks. CNNs are modeled after the structure of the visual cortex in the brain, which is composed of small, local regions that are sensitive to specific features in an image.

CNNs consist of several layers, including an input layer, one or more convolutional layers, one or more pooling layers, and one or more fully connected layers. The input layer receives the image data, which is typically represented as a multi-dimensional array of pixel values. In the convolutional layers, a set of filters is applied to the input image to extract features from it. These filters are typically small and local and are designed to be sensitive to specific features in the image, such as edges or textures. The filters are moved across the image in small steps, and the output of each step is called a feature map. The pooling layers are used to reduce the spatial size of the feature maps and to make the feature maps more robust to small changes in the image. This is done by applying a pooling operation, such as max pooling, to small regions of the feature maps. The operation returns the maximum value in each region, and the result is a new, smaller feature map. After the convolutional and pooling layers, one or more fully connected layers are used to make a prediction based on the features extracted from the image. The fully connected layers are similar to traditional neural networks and are used to make the final decision based on the features learned by the previous layers.

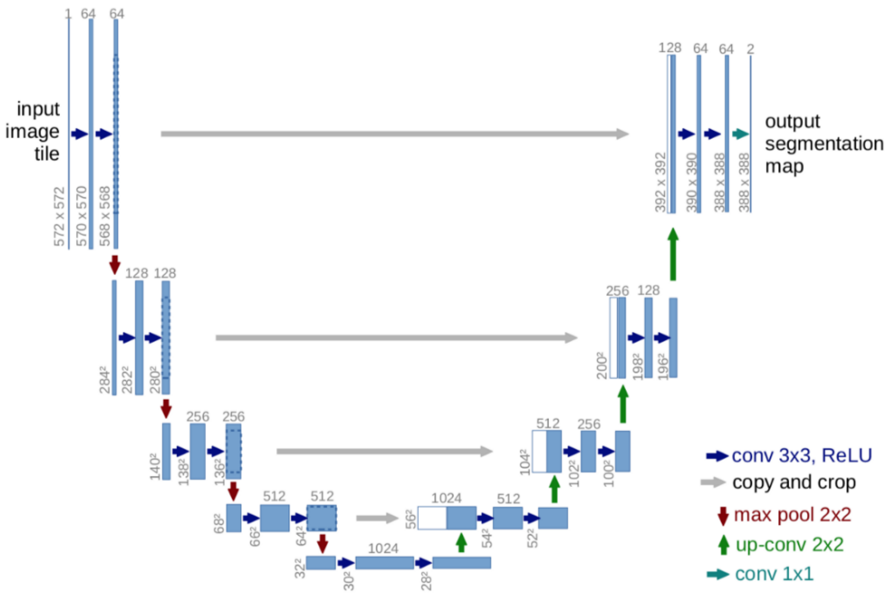
CNNs are particularly well-suited to image recognition and processing tasks because they can learn to recognize features in images that are invariant to small translations, rotations, and other transformations. This allows them to perform well on images that are not perfectly aligned or that have small variations.



CNN architecture

Emergence of U-Nets:

A U-Net is a type of convolutional neural network (CNN) that is commonly used for image segmentation tasks. The U-Net architecture is composed of two parts: the contracting path and the expanding path. The contracting path is a sequence of convolutional and pooling layers that extract features from the input image and reduce the spatial resolution of the feature maps. The expanding path is a sequence of transposed convolutional layers that increase the spatial resolution of the feature maps and use the features learned by the contracting path to make a pixel-wise prediction of the output image. One of the main features of the U-Net architecture is the use of skip connections between the contracting and expanding paths. These connections bypass the pooling layers and allow information to flow directly from the contracting path to the expanding path, allowing the U-Net to make use of both low-level and high-level features. The U-Net architecture is particularly well-suited for image segmentation tasks because it can make use of both global and local information in the image. The contracting path allows the network to extract features from the entire image, while the expanding path allows the network to make pixel-wise predictions. The skip connections also allow the network to make use of both low-level and high-level features to make its predictions.



UNet architecture

Application of deep learning and U-Nets in the medical field:

Deep learning has been applied in a variety of ways in the medical field. Some examples include:

1. Medical image analysis: Deep learning algorithms have been used to analyze medical images, such as X-rays, CT scans, and MRIs, for tasks such as image segmentation, object detection, and image registration. These algorithms can be used to automatically detect tumors, identify specific structures in an image, or align different images of the same patient.

2. Diagnosis and treatment planning: Deep learning algorithms have been used to support the diagnosis and treatment planning for a variety of medical conditions. For example, deep learning algorithms have been used to predict the likelihood of developing certain diseases, such as Alzheimer's or cancer, based on imaging data or electronic health records.

3. Medical natural language processing: Deep learning algorithms have been used to process and analyze large amounts of unstructured medical text data, such as electronic health records and scientific literature. These algorithms can be used to extract information from the text, such as identifying specific medical conditions or extracting patient data.

4. Medical signal processing: Deep learning algorithms have been applied to process medical signals such as ECG, EEG, etc. The algorithms can be used to identify abnormal patterns in the signals, such as arrhythmias in an ECG, and can assist in the diagnosis of certain diseases.

5. Robotics: Deep learning algorithms have been used in the development of medical robots, such as surgical robots, that can assist surgeons in performing procedures.

Deep learning has the potential to revolutionize the medical field by providing more accurate and efficient diagnostic tools, better treatment planning, and improved patient outcomes. However, deep learning in the medical field is still in its early stages and more research is needed to fully explore its potential and address some of the challenges facing its adoption in clinical practice.

As for U-Nets, it has been applied in various medical imaging tasks. Some examples of how U-Nets have been used in the medical field include:

1. Image segmentation: U-Nets have been used to perform image segmentation tasks, such as segmenting tumors, organs, and other structures in medical images. This can be useful for tasks such as radiation therapy planning, surgical planning, and lesion quantification.
2. Medical image registration: U-Nets have been used to align different images of the same patient, such as pre- and post-operative images, to improve the accuracy of image-based diagnosis and treatment planning.
3. Medical image synthesis: U-Nets have been used to generate synthetic medical images, such as CT scans, from other modalities, such as MRI, to improve the accuracy of diagnosis and treatment planning.
4. Cell Segmentation: U-Nets have been used to segment and classify cells in microscopy images to assist with medical research and diagnosis.
5. 3D medical imaging: U-Nets have been extended to work with 3D medical images, such as CT scans, and can be used to segment and classify structures in 3D.
6. Medical data augmentation: U-Nets have been used to generate new medical images from existing ones, by applying different transforms such as rotations, translations or adding noise, to improve the performance and robustness of the models.

U-Nets are widely used in the medical field due to their ability to handle large amounts of data, their robustness to small variations, and their ability to extract features from both the global and local context of an image. Its architecture and skip connections make it a powerful tool for image segmentation, registration and synthesis tasks in the medical field.

Data Collection and Pre-processing

Data Collection

The original dataset used for this project is the open-source ATLAS 2.0 dataset, which is publicly available in a preprocessed format (standardized to MNI-152 space) on INDI (International Neuroimaging Data-Sharing Initiative), a free platform for neuroimaging data sharing.

The data is encrypted using OpenSSL and is available for download directly from NITRC (Neuroimaging Informatics Tools and Resources Clearinghouse) or from an Amazon Web Services S3 bucket. And In order to obtain the password needed to access the data, we had to agree to the Terms of Use specified on the website and complete a brief Google form for each member.

After decrypting the data, the ATLAS 2.0 includes:

- 955 T1-weighted MRI scans, divided into a training dataset (n=655 T1w MRIs with manually segmented lesion masks) and a test dataset (n=300 T1w MRIs only; lesion masks not released)
- MNI152 standard-space T1-weighted average structural template image
- Two .csv files containing lesion and scanner metadata

The data is maintained in a BIDS (Brain Imaging Data Structure) format. There are 33 cohorts in the training and testing datasets, and within each cohort folder are individual subject folders. The following naming convention was used: sub-r***s*** where r*** represents the research cohort number and s*** represents the subject number. All data are cross-sectional and from a single timepoint, so they all are denoted with “ses-1”. Native space images are labeled as “space-orig” while images normalized to the MNI-152 template are labeled as “space-MNI152Nlin2009aSym”.

Following BIDS conventions, a lesion mask in native space would be named as such: “sub-r***s***_ses-1_space-orig_label-L_desc-T1lesion_mask.nii.gz” and the corresponding T1w MRI would be named as “sub-r***s***_ses-1_space-orig_T1w.nii.gz.” So, the T1w MRI and lesion mask in MNI space are noted as: “sub-r***s***_ses-1_space-MNI152NLin2009aSym_T1w.nii.gz” and “sub-r***s***_ses-1_space-MNI152NLin2009aSym_label-L_desc-T1lesion_mask.nii.gz”, respectively.

Data Pre-processing

Once the data was downloaded, decrypted and collected, it was time to pre-process the data before exploiting it. Basically, the data came in a NIFTI format (.nii file) to respect the BIDS format.

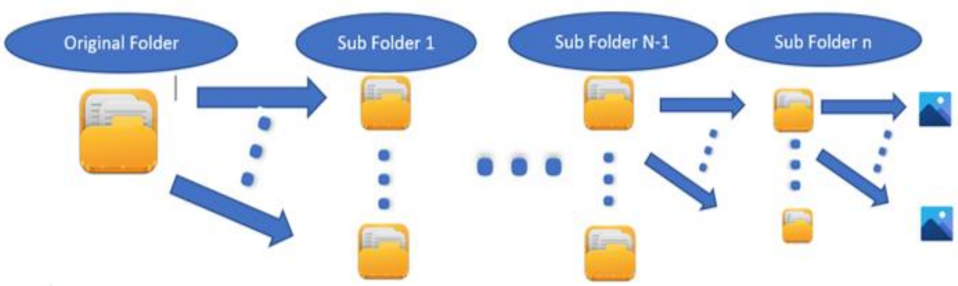
So, for our models to work properly, we decided that the input hyper-meters have to 2D slices of the 3D brain MRIs and masks, so that the models can learn easily and efficiently.

To slice the samples of our data, we developed a python script using the NiBabel.

The NiBabel python library is mainly used for neuroimaging data, that gives access to a variety of imaging formats, with a particular focus on providing a common interface to the various volumetric formats produced by scanners and used in common neuroimaging toolkits :

- NIfTI-1
- NIfTI-2
- SPM Analyz
- FreeSurfer .mgh/.mgz files
- Philips PAR/REC
- Siemens ECAT
- DICOM (limited support)

So, our python script contains 2 main functions to slice a nii file and save it as a png 2D image.
Since our data is organized following BIDS conventions as mentioned before, meaning there are at least 3 sub-folders we need to open before accessing the raw nii files, as the following diagram shows :



The function **open_folders()** that takes 2 arguments : folder_path to check if the input path is a folder/sub-folder itself or a nii file, and if it's indeed a directory, the functions adds this item to the input path and recursively calls itself again with the new path. This operation goes on until we finally get to a directory with no sub-folders, meaning only raw data. Also, the function checks if the present files in this directory are really NIFTI files by checking the extension (.nii).

```
#%% Open all the folders in the ATLAS_R2.0 directory
### and save the path of all the nii files in a list

global nii_files_list

def open_folders(folder_path,path_list):
    # Check if the path is a directory or an nii file
    if os.path.isdir(folder_path):
        # Get the list of items in the directory
        items = os.listdir(folder_path)
        for item in items:
            item_path = os.path.join(folder_path, item)
            # Recursively call the function for the item
            open_folders(item_path,path_list)

    elif os.path.basename(folder_path).split('.')[1] == 'nii':
        # Add the path of the nii file to the list
        path_list.append(folder_path)

    return path_list
```

The second function, and the most important one, **convert_nii2png()** slices a nii file by iterating through the slices and saving each sample as a png image. It takes into argument, the input_path to read and load the NIFTI data, and the output-path to save the slices.

```
### Convert a nii file to a png image
def convert_nii2png(input_path,output_path):
    image_array = nibabel.load(input_path).get_data()
    nx, ny, nz = image_array.shape

    total_slices = image_array.shape[2]
    slice_counter = 0
    # Iterate through slices
    for current_slice in range(0, total_slices):
        # Alternate slices and save as png
        if (slice_counter % 1) == 0:
            data = image_array[:, :, current_slice]
            if (slice_counter % 1) == 0:
                image_filename = "slice_" + "{:0>3}".format(str(current_slice+1))+ ".png"
                #imageio.imwrite(image_filename, data.astype(np.uint8))
                imageio.imwrite(image_filename, data)
                src = image_filename
                shutil.move(src, output_path)
                slice_counter += 1
```

Problems and solutions:

-No samples of healthy brains in the dataset

We created 2d slices from the 3d images, this allowed us to do more data augmentation and introduce healthy images in your training set.

-The images in the dataset are preprocessed, meaning we cannot add more images from other sources because of the eventual biases that can occur.

We added data augmentation layers in the entry of our U-net model to simulate new and diverse samples with each epoch.

-Some images and especially the masks had some noise that was messing up the predictions (black pixel values not 0)

We used thresholding on the masks in order to force the values of the pixels to zero.

-The white representing the brains had different intensities (pixel values).

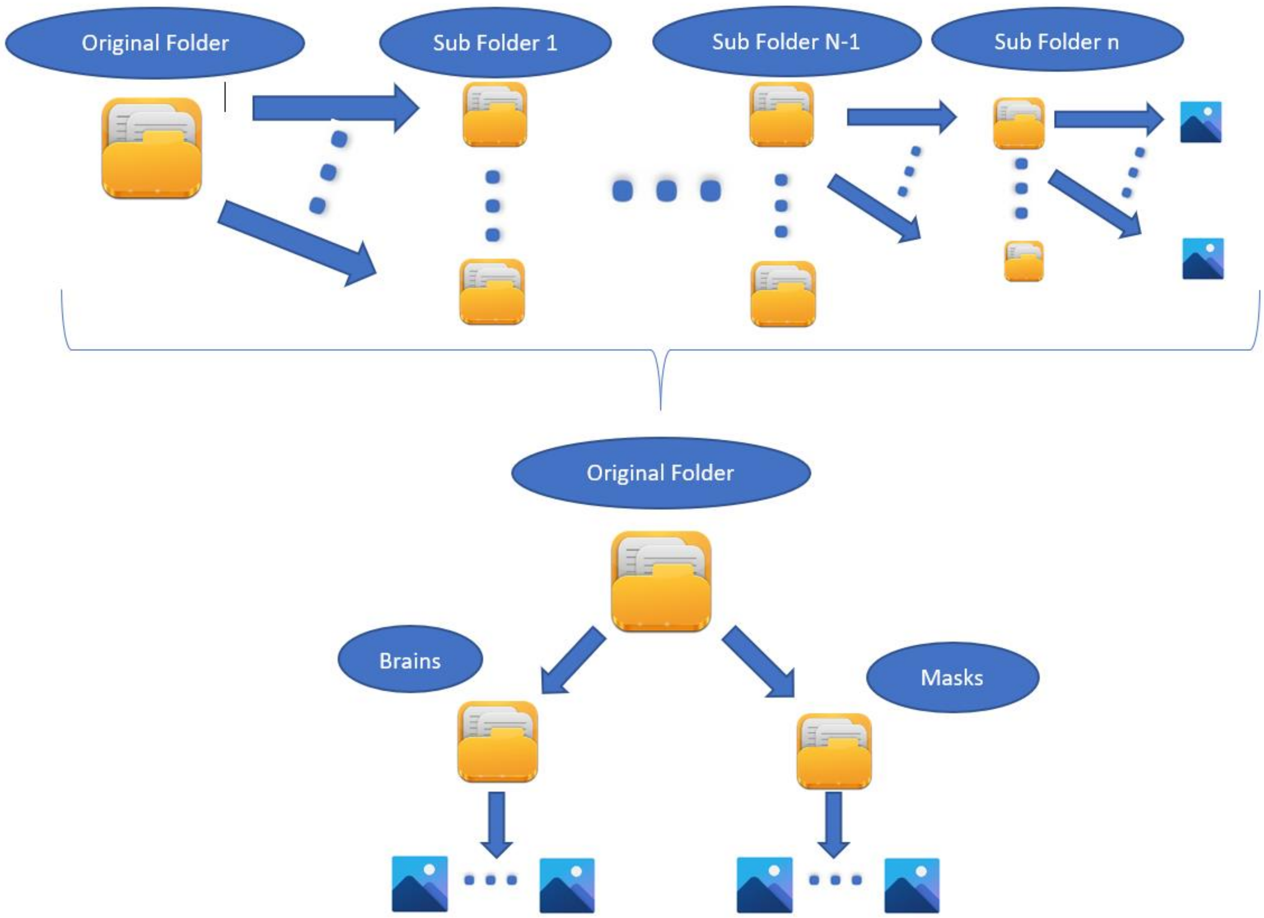
Data Cleaning and formatting:

After having our data collected and pre-processed it is time for some deep data cleaning and formatting. To begin with, let's talk about formatting that was needed for the project.

For this specific project we needed to have a one-to-one correspondence between the dataset to able to train our model to take a brain as the input and have the mask as expected out to be trained based on that.

The original pre-processed data came as the raw data which means every brain was put specific file subfile. This type of formatting doesn't really go with what the model needs; therefore, formatting was needed to collect every brain and every mask with a one to one correspondence without any mixes between masks and brains order.

This diagram will show how the transformation was done:

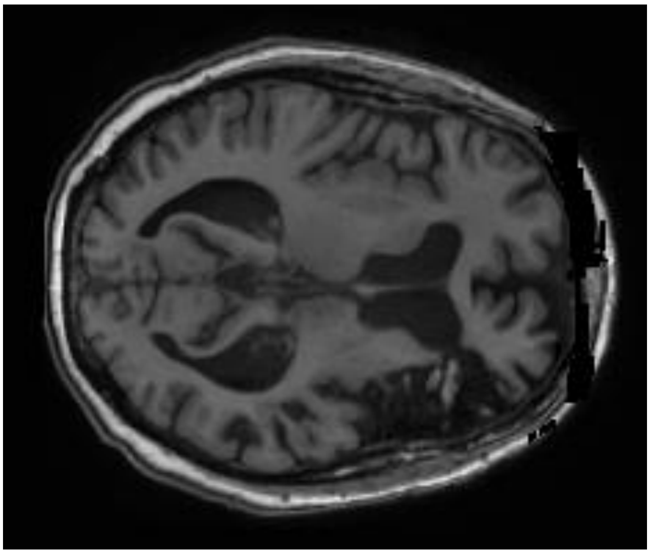


With this format in place the model is ready to take inputs to be trained.

P.S: All codes can be found in the attached notebook, **formatting_and_cleaning_brain_project.ipynb**

Now that formatting of the files is covered let's talk about the formatting of the images themselves.

We received the original images as this format:



As we can see the brain is shifting to the right with a rectangular ration.

For the data format as we took several formats and that can be seen in our approaches as manipulating that data was one of our main ways to tackle different approaches.

Approaches:

All the approaches' models at their core are relatively the same. However, their main difference is the way the model was fed for them.

Beginning to talk about the Core model in itself:

The U-Net model is a convolutional neural network (CNN) architecture designed for image segmentation tasks. It is particularly useful for biomedical image segmentation, as it is able to effectively handle the small object sizes, high variability and low contrast often found in these images. The U-Net architecture is a fully convolutional network that consists of a contracting path to capture context, and a symmetric expanding path that enables precise localization. The contracting path is a typical CNN, while the expanding path uses upsampling layers to increase the spatial resolution of the feature maps. The U-Net model is trained end-to-end to predict a segmentation mask for a given input image.

U-Net is used in health imaging because it has been shown to be effective in many biomedical image segmentation tasks, such as cell segmentation, neuron segmentation, and organ segmentation. It is also commonly used in medical image analysis such as in the case of segmentation of CT and MRI scans for cancer diagnosis, treatment planning and follow-up evaluations.

Now that we have covered the core of our model let's see the different approaches we took.

We will go in chronological order.

<u>Environment:</u> We used for our project google collab and we utilized its GPU.

First approach: (basic_model)

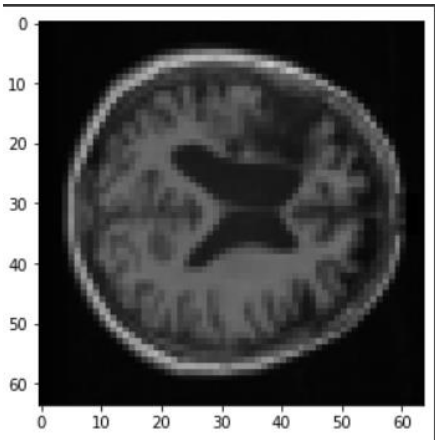
For the first approach we worked with the raw data after building our model meaning:

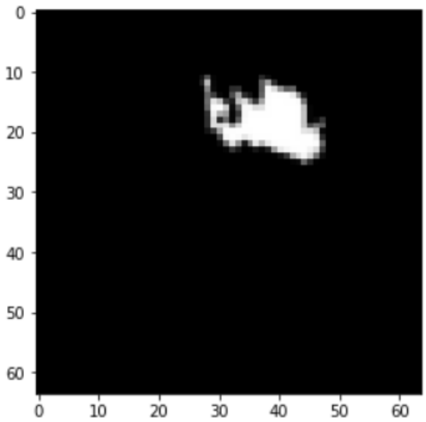
- Using all the brain images from 0 TO 189.
- All of them had the size of (197,233).

Since we can't use the images without having a square ration, we needed to change the size of our brains and masks and for first approach we went with (64,64) Dimensions.

Here is what our data looked like:

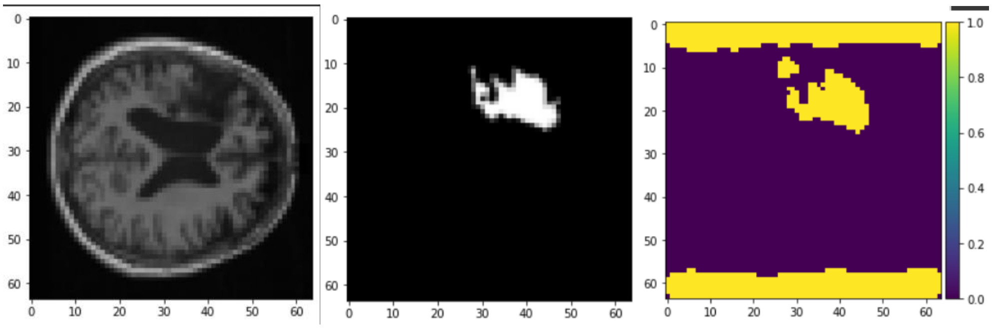
Brain Mask





This is the brain and its correspondent mask.

Result of the first model:



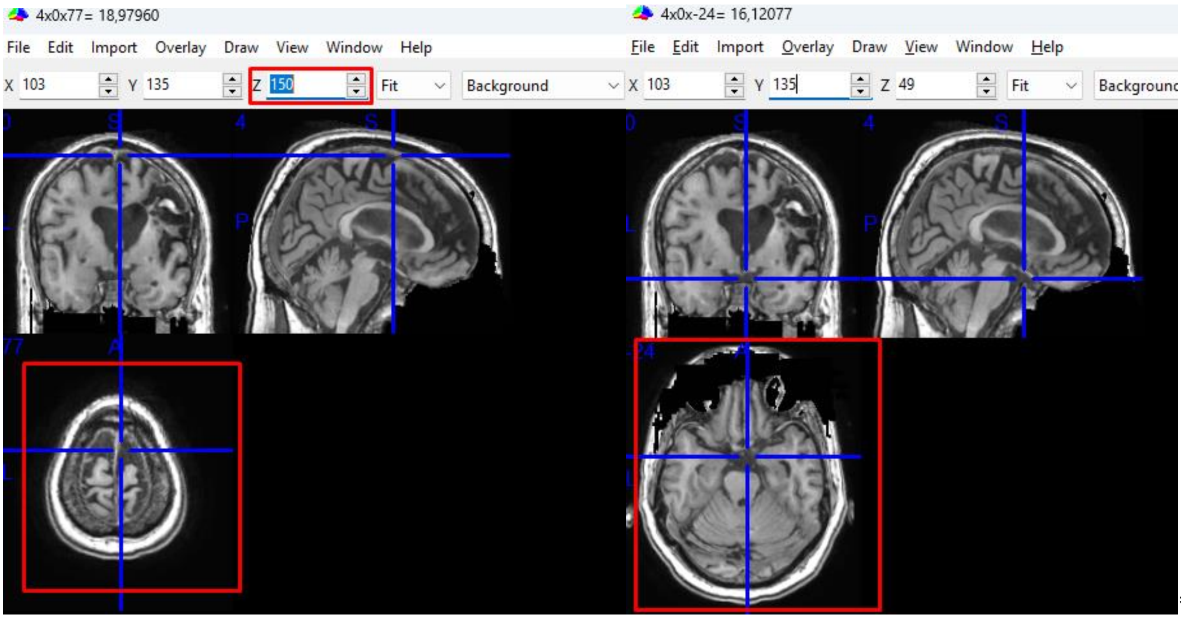
As we can see our prediction model detects some of the tumor however it is also predicting a lot of noises.

Second approach: (Second_model)

Moving forward we decided to overcome this challenge by taking several steps:

- Reducing the amount of images per brain from 0 -> 189 to 50 -> 150
- Having a higher quality image from 64 to 128
- Adding padding to the original brain on the sides to have a square dimension for us not to lose any information while downsizing.
- Using thresholding on masks to have only black and weight colors

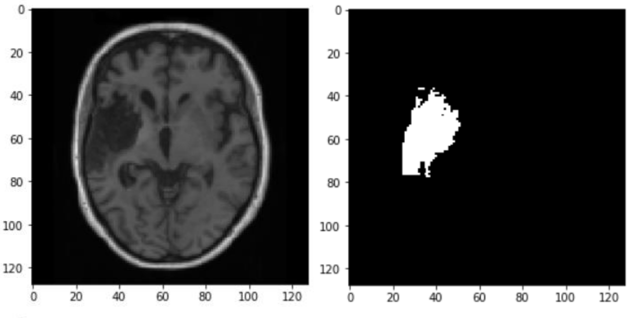
Since we needed to use higher images and more computing power, we also started to use google collab pro as our environment to work on.



*

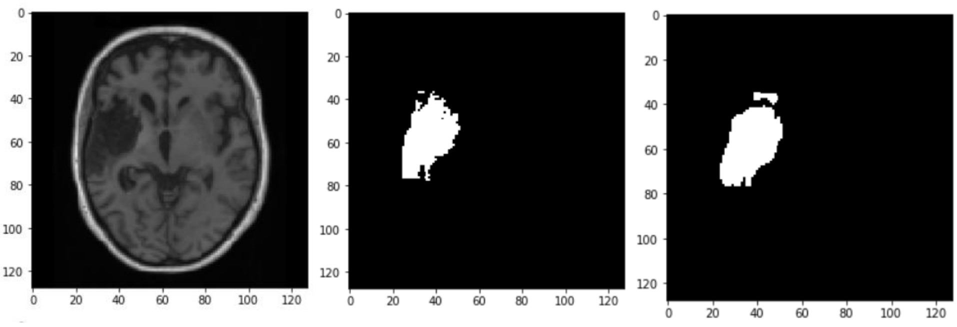
Example of the input for the model :

Brain - Mask



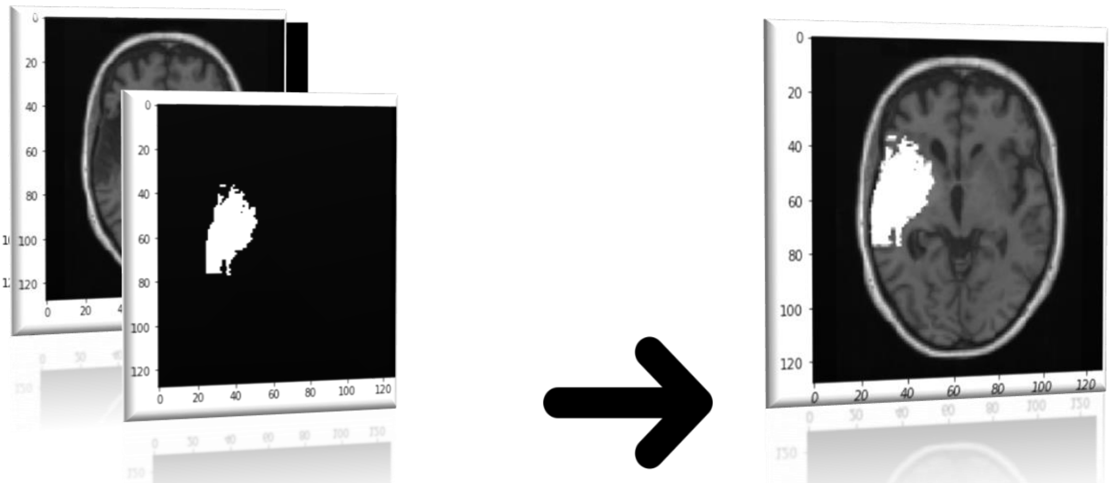
We trained the model for 7 epochs (could not do more as be keep getting disconnected and banned from GPUs)

And got the following results: (demo at the end)



Third approach: (Third Model)

For this approach working with data was a bit different since the masks has more size than the brains because for the third approach we decided to see what would happen if the output layer was actually 2 layers on top of one another.



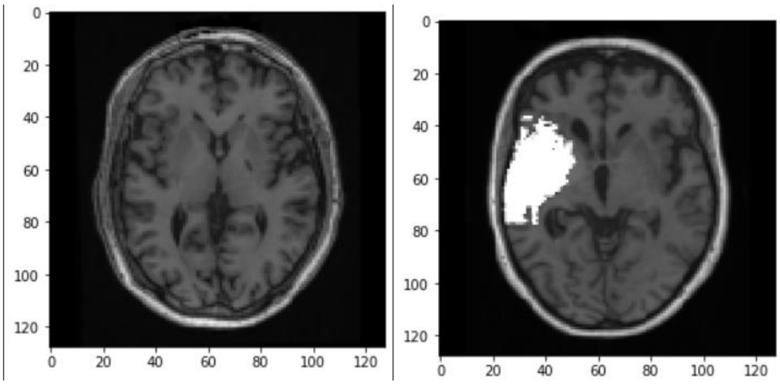
For the third approach I worked with the following parameters:

- Reducing the amount of images per brain from 0 -> 189 to 50 -> 150
- Having a higher quality image from 64 to 128
- Adding padding to the original brain on the sides to have a square dimension for us not to lose any information while downsizing.

- The mask will be considered of 2 levels first level of the original brain and the second level the mask as well.

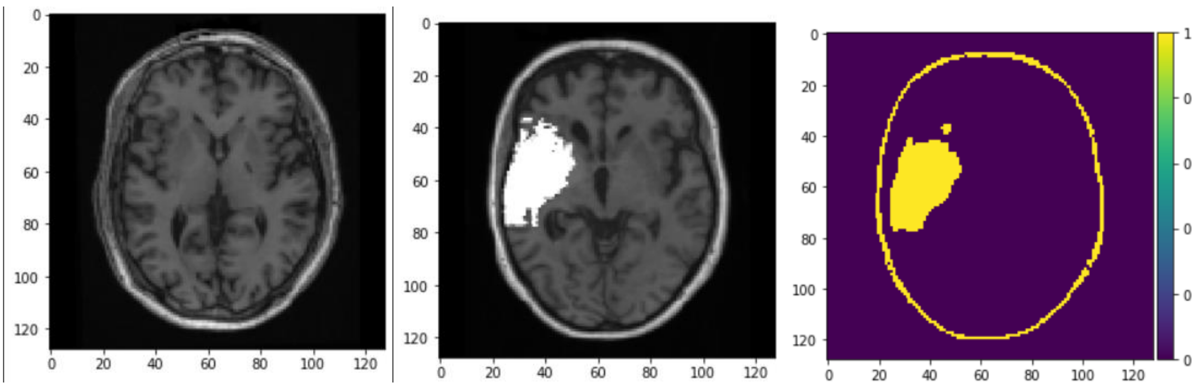
Since we needed to use higher images and more computing power, we also started to use google collab pro as our environment to work on.

Brain   Mask



For this model we got the following result:

Brain   Mask   Predicted



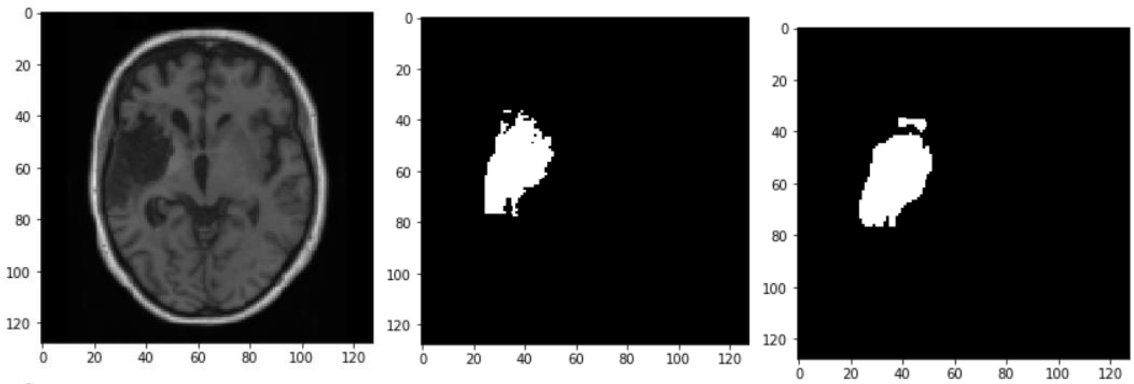
Comparison:

Now let's do some Comparision between the 2 models. I will be comparing several brains with how each model behaved:

1- <u>First Brain:</u>

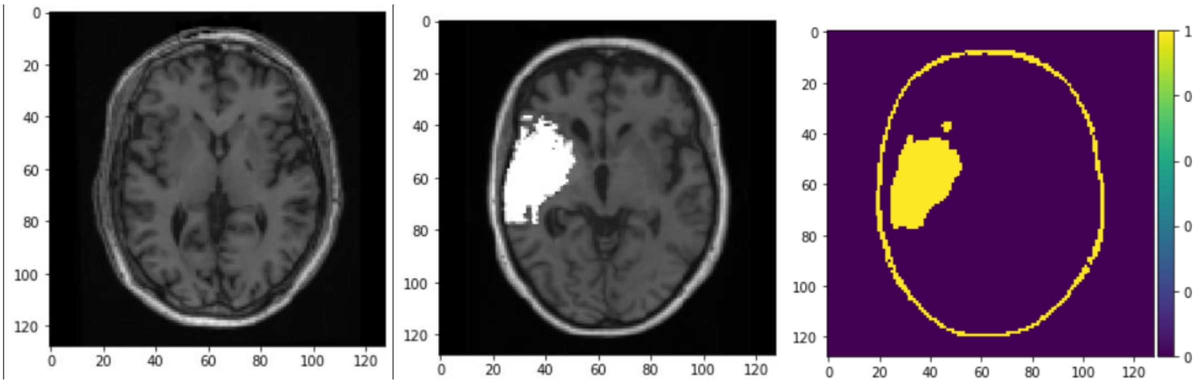
   a. <u>Model 2:</u>

brain   Mask   Predicted



b. Model3:

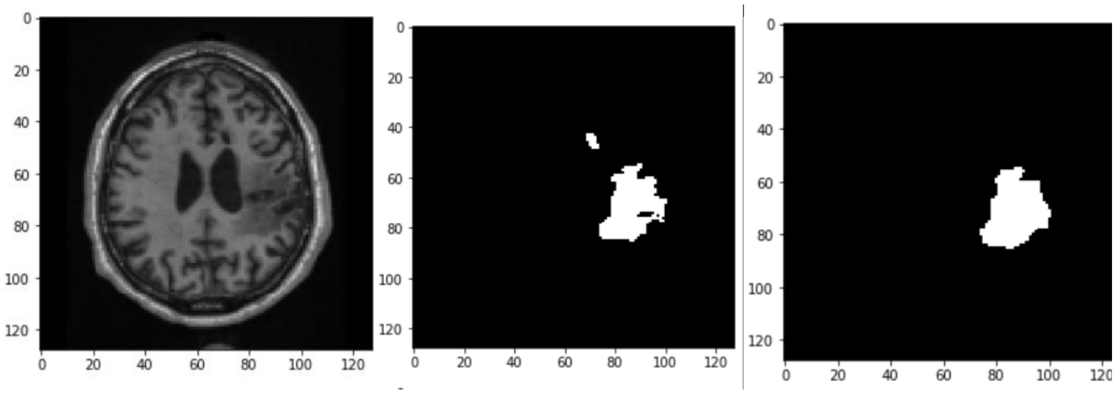
brain Mask Predicted



2- Second Brain:

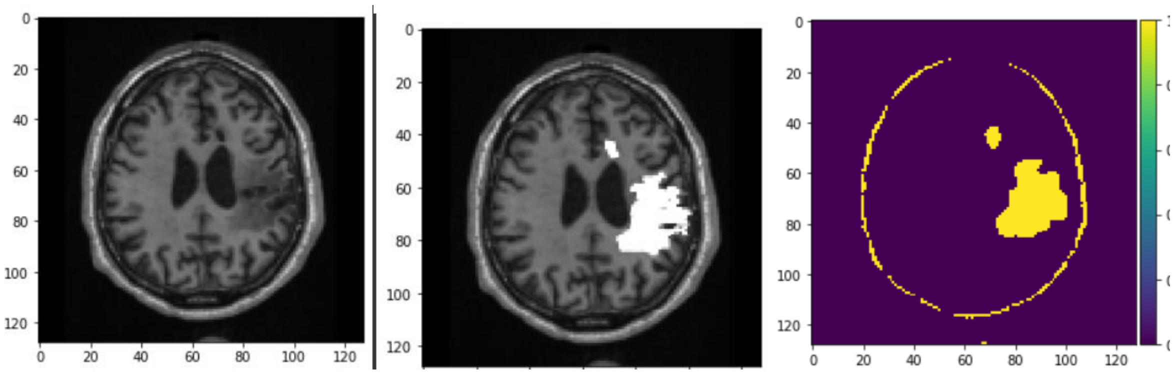
a-Model 2:

brain Mask Predicted



b-Model 3:

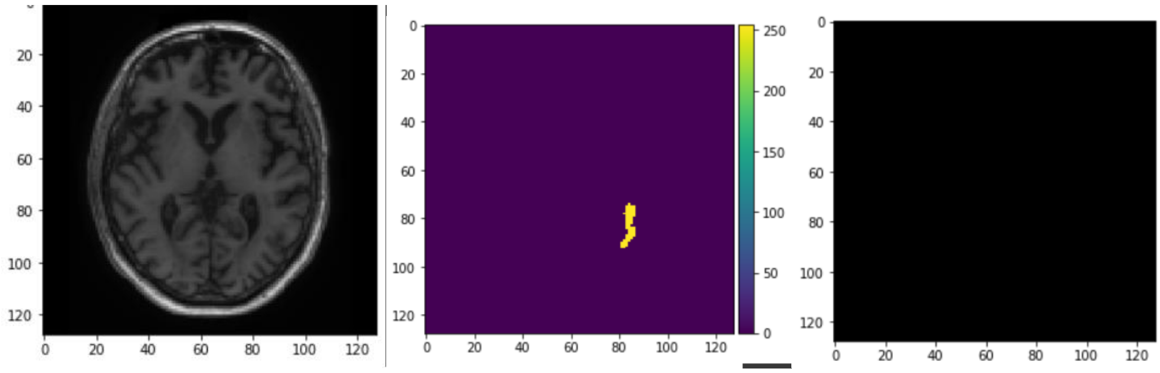
brain Mask Predicted



3-Third Model :

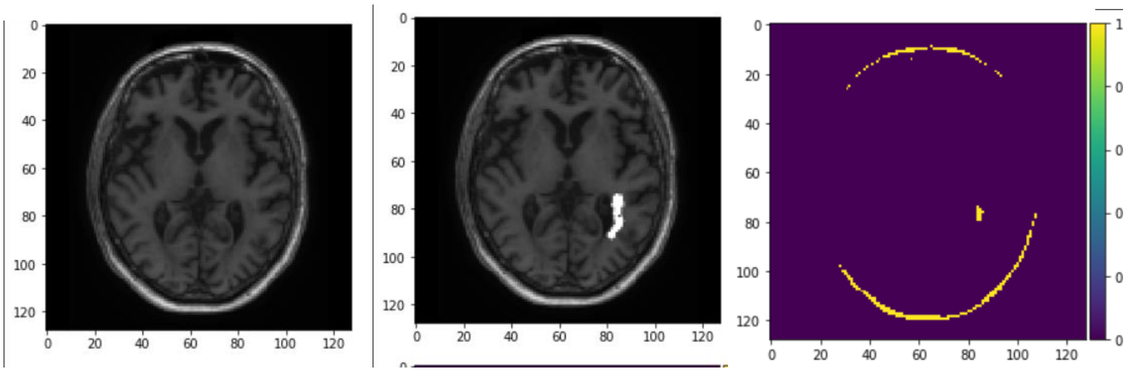
A-Model 2:

brain Mask Predicted



B-Model 3:

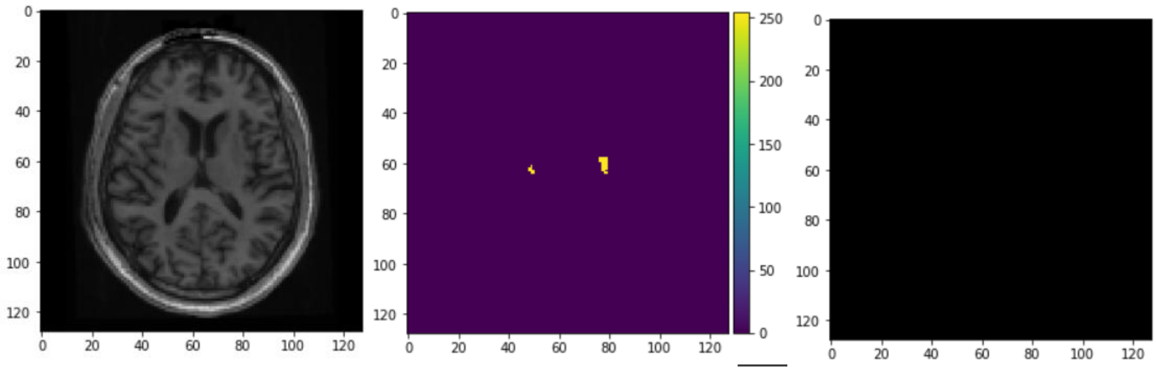
brain Mask Predicted



4- Forth model :

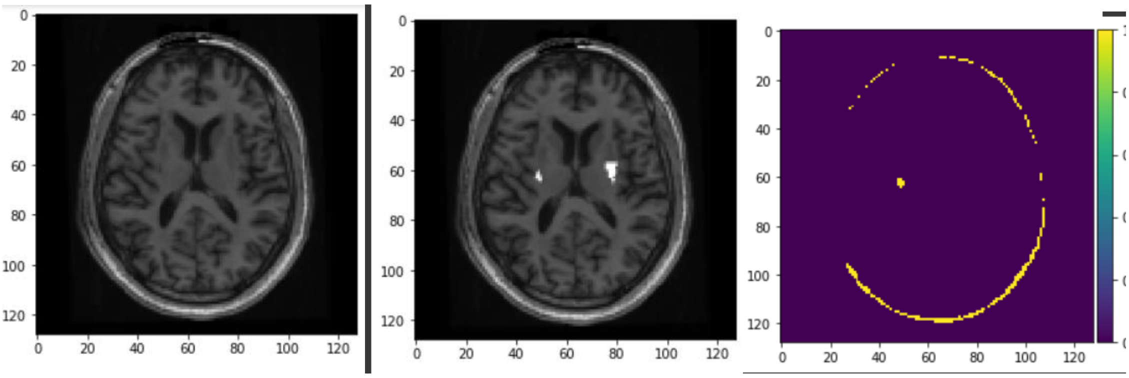
A-Model 2:

brain Mask Predicted



B-Model 3:

brain Mask Predicted



Discussions

The two models performed decently well in different ways: the 2nd model don't predict the outline of the brain giving its prediction more accuracy, while the 3rd model is generally more accurate with the strokes and is able to detect smaller strokes more frequently.

Critique

- These approaches are definitely steps in the right direction but the current model still lack accuracy in order to be considered for deployment.
- These models were not tester on real world example due to the confidentiality of patients date.

future work

This work can easily be improving with better thresholding methods applied to the 3rd model to remove the prediction of the brain's skull as well as using better machines with more RAM and GPUs for better fine tuning of the models.