Classification of Stroke Blood Clot Origin

Supervised By:

Dr.Srilatha Chebrolu

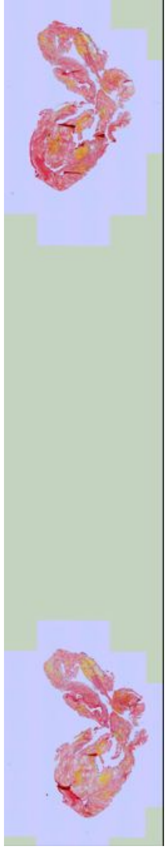
Presented By:

411958 - P. Surya
411950 - M. Raghavendra Rao
411918 - E. Nikhil Sai

Problem Statement

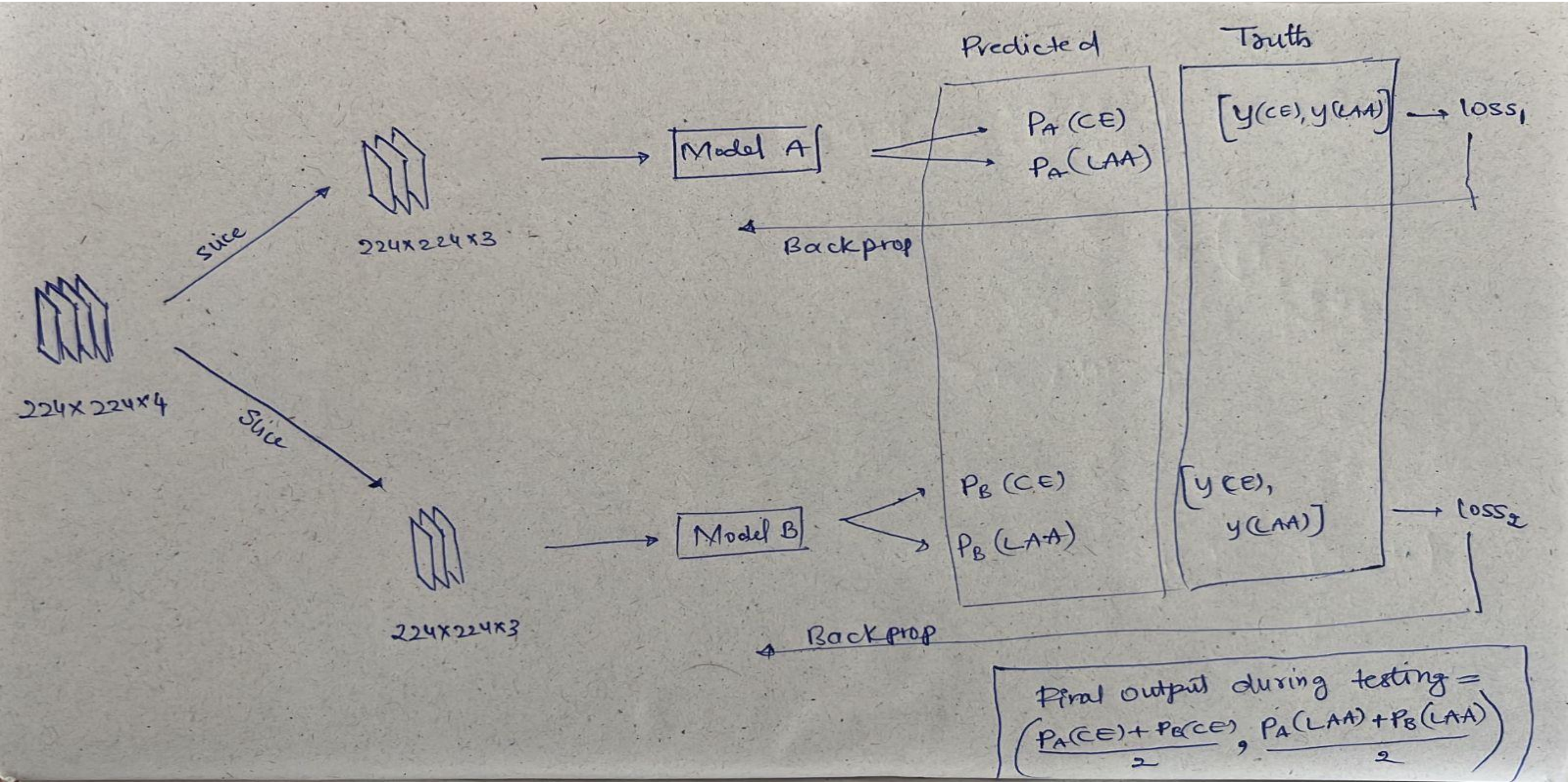
- To identify the blood clot origin in ischemic stroke using Whole slide digital pathology images.
- We need to build a model that differentiates between the two major acute ischemic stroke(AIS) etiology subtypes: 'Cardioembolic' (CE) and 'Large Artery Atherosclerosis' (LAA).

A sample digital pathology image labelled as ‘Large Artery Atherosclerosis’.



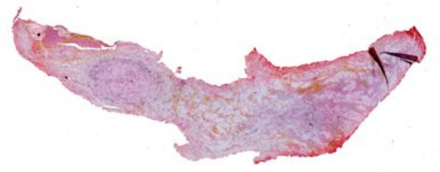
A sample digital pathology image labelled as ‘Cardioembolic’ .

Previous work



Methodology

- **Dataset :** The dataset contains 754 high resolution whole slide images each belonging to the blood clot origin cause, out of which 547 images belong to class CE, and 207 images belong to class LAA.



- **Data Preprocessing** : The data preprocessing is divided into following steps:
    i. Tiling
    ii. Filtering
    iii. Color normalization
    iv. Augmentation
    v. Rescaling

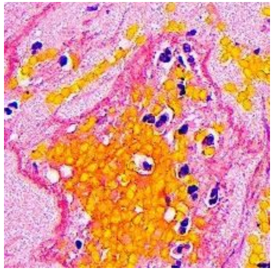
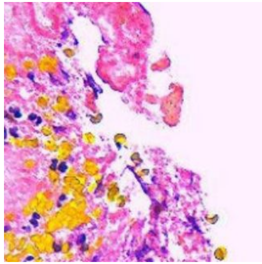
1. **Tiling** : Each WSI is tiled into multiple patches each of shape 512 x 512, using the openslide deep zoom generator. To handle the class imbalance, atmost

   6 tiles from a image that belong to class CE and atmost

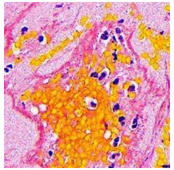
   22 tiles that belong to class LAA have been selected.

   Finally, the training set consists of

   2569 images belonging to class CE and 2302 images belonging to class LAA.

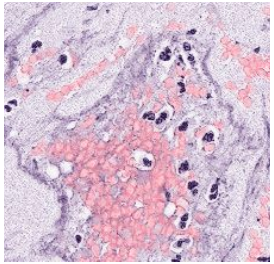



2. **Filtering**: The tiles with white or mostly white backgrounds are discarded and also the tiles that do not carry important information are ignored. The tiles that satisfy the threshold conditions of mean and standard deviations are filtered and added to the final training set. Tiles which are dark and has higher standard deviation are considered into the training set. For this dataset, the mean threshold is set to 180 and standard deviation threshold is set to 50. This image for example is eliminated as the mean and standard deviation values are 208 and 59 respectively



3. **Color normalization**:

- There will be differences in color intensity and shading in the images, which might be possible due to various factors such as slide preparation,staining methods and imaging device variability.
- So images need to be color normalized, so they do not affect the analysis and interpretation of the images.
- Output: Color normalized image , Hematoxylin and eosin
- These color normalized images are added to the final training set.



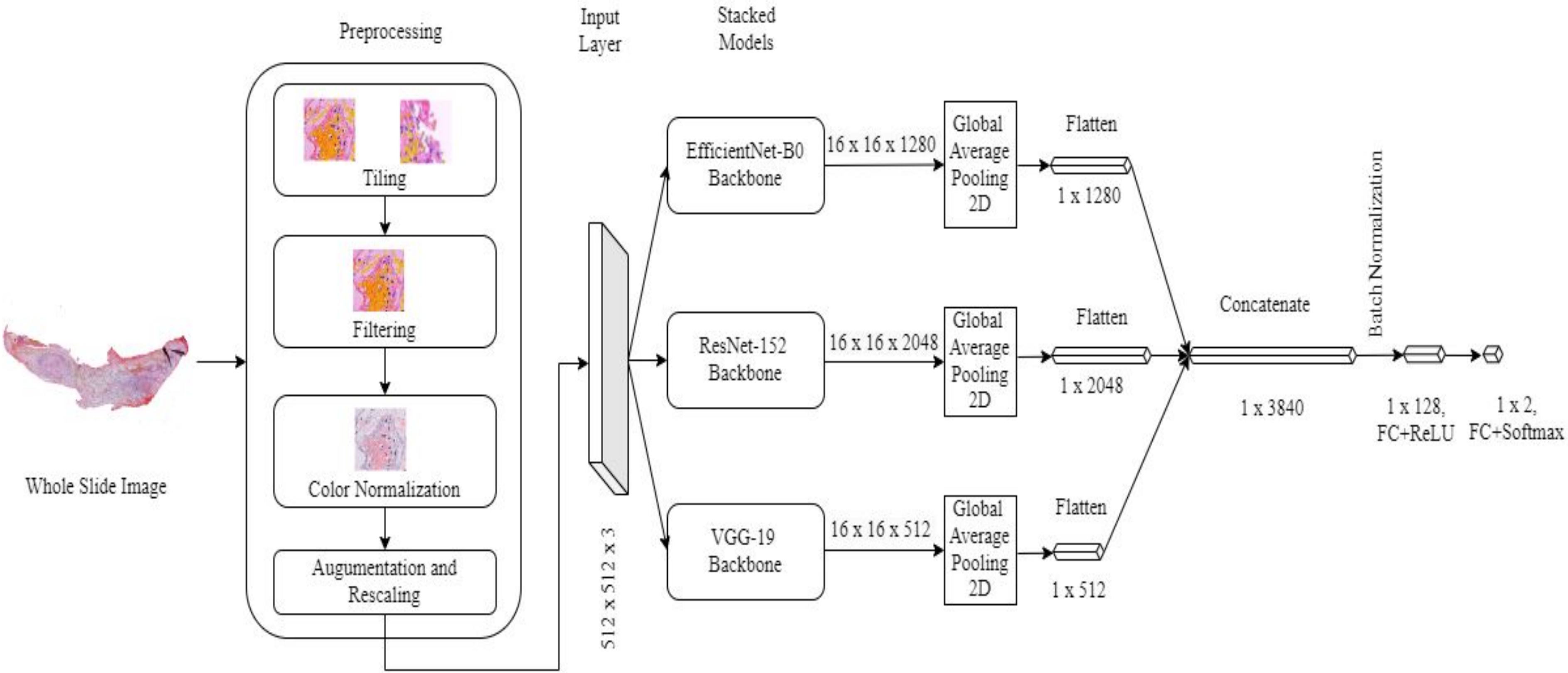
4. **Augmentation** : Data augmentation techniques like rotating, increasing the brightness, zooming, vertical and horizontal flips, etc are applied to the tiles.

5. **Rescaling** : The images are rescaled by a factor of 255 before passing into the model.

- **EfficientNet-B0, ResNet-152 and VGG-19 stacked model:**
  - The three backbones of the CNN architectures are pre-trained on ImageNet dataset and fine-tuned on out dataset.
  - The first layer of the model accepts RGB images of shape $512 * 512$. The feature extractor outputs from the three backbone models EfficientNet-B0, ResNet-152 and VGG19 are passed through a global average pooling layer. These outputs are individually flattened and concatenated to obtain a single output of shape $1 * 3840$.
  - Batch normalization layer is used after flattening and concatenating the feature extraction outputs. This is followed by a hidden dense layer having 128 neurons with ReLU activation function. The output layer of the model has two neurons with softmax activation function and provides the probabilities of each class label. The total parameters of this implementation is 82,952,165, out of which 499,586 are trainable.

Layer	Output shape	Parameters
Input Layer	512*512*3	0
EfficientNet-B0 feature extractor	16*16*1280	4,049,571
ResNet-152 feature extractor	16*16*2048	58,370,944
VGG19 feature extractor	16*16*512	20,024,384
Concatenation	1*3048	0
Batch Normalization	1*3048	15,360
Dense	1*128	491,648
Dense	1*2	258

Model architecture :



Hyper parameters

- The shape of the images in the training set is 512*512.
- Adam is the optimizer function with learning rate set to 0.001 and the loss function used for calculating the error during back propagation is categorical cross entropy.
- The batch size of the training set is set to 32 and the model is trained for 40 training steps.
- Categorical cross entropy:

$$\frac{-1}{R}\sum_{i=1}^{R} a_i log\hat{a}_i + (1-a_i)log(1-\hat{a}_i)$$

- Weighted multi-class log loss: This loss is used for the evaluation of the model after training.

$$\frac{-\sum_{i=1}^{C} W_i \sum_{j=1}^{R_i} \frac{y_{ij}}{R_i} ln\hat{y}_{ij}}{\sum_{i=1}^{C} W_i}$$

Results

Model	Input size	Epochs	Batch size	Loss
CNN	224	30	32	0.74833
EfficientNet-B0	224	30	32	0.73246
ResNet-152	224	30	32	0.73490
VGG16	224	30	32	0.82692
EfficientNet-B0, VGG19 stacked	224	30	32	0.78061
EfficientNet-B0, ResNet-152 stacked	224	30	32	0.74695
EfficientNet-B0	512	60	16	0.69366
EfficientNet-B0	512	40	32	0.69332
ResNet-152	512	60	16	0.69597
ResNet-152	512	40	32	0.69854
VGG19	512	40	32	0.70645
ResNet-152, VGG19 stacked	512	40	32	0.69792
ResNet-152, VGG19 stacked	512	60	16	0.69943
EfficientNet-B0, VGG19 stacked	512	40	32	0.75251
EfficientNet-B0, VGG19 stacked	512	60	16	0.73355
EfficientNet-B0, ResNet-152 stacked	512	40	32	0.69484
EfficientNet-B0, ResNet-152 stacked	512	60	16	0.71081
EfficientNet-B0, VGG19 ResNet-152 stacked	512	40	32	0.69312
EfficientNet-B0, VGG19 ResNet-152 stacked	512	60	16	0.71019

Score calculation

Weighted multi-class logarithmic loss

$$\frac{-\sum_{i=1}^{C} W_i \sum_{j=1}^{R_i} \frac{y_{ij}}{R_i} ln\hat{y}_{ij}}{\sum_{i=1}^{C} W_i}$$

where C, R represents the number of classes, and number of records respectively
yij denotes whether image i belongs to class j in the training set
ˆyij is the predicted probability of image i belonging to class j
and Wi denotes the weight of
class i.