# Sprint 3:

# RSNA-MICCAI Brain Tumor Radiogenomic Classification

Team members: Zhaowen Zhou
Rahaf Alharbey
Jiawei Zhao
Shuyi Fan

# Contents

1. Data Preparation

2. Model 1: ResNet Model

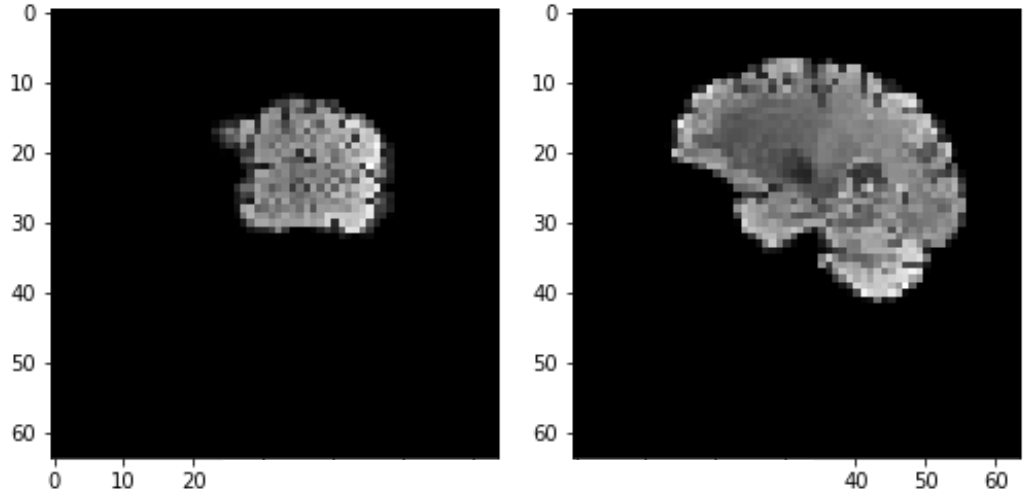
3. Model 2: EfficientNet 3D

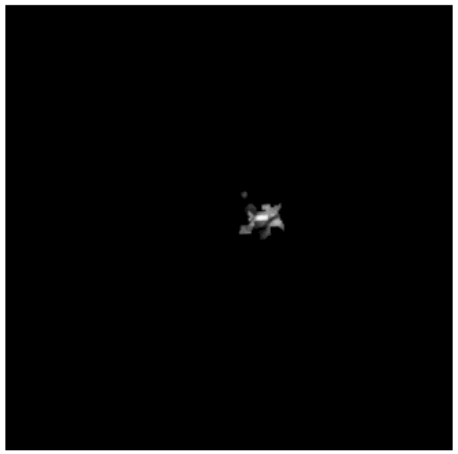
4. Next Step

# Data Preparation

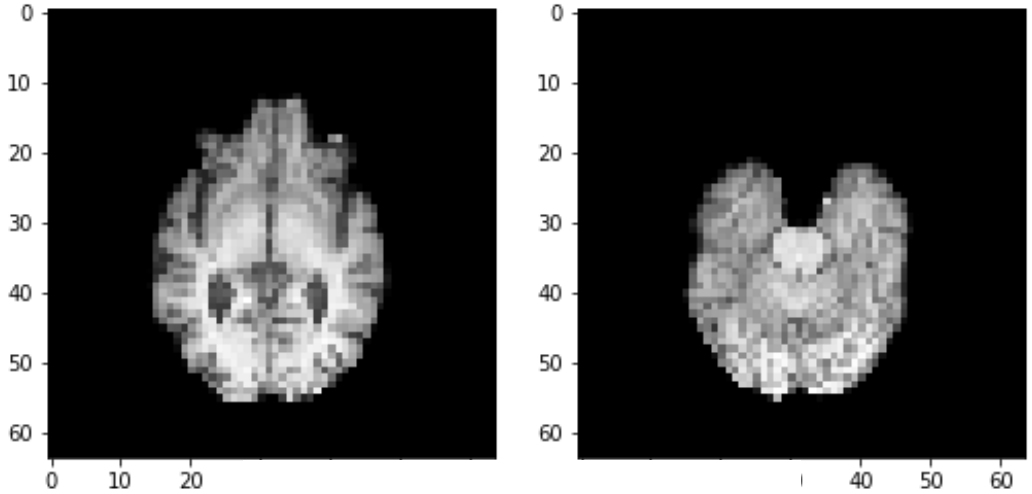
# Input Data

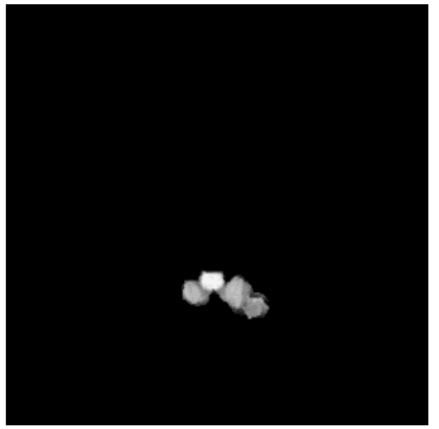
## FLAIR Picture



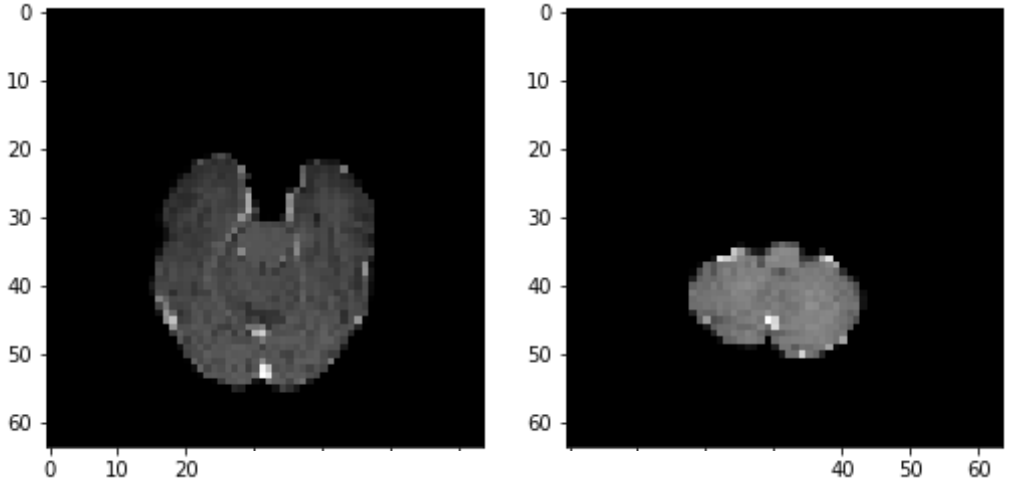
## T1w Picture

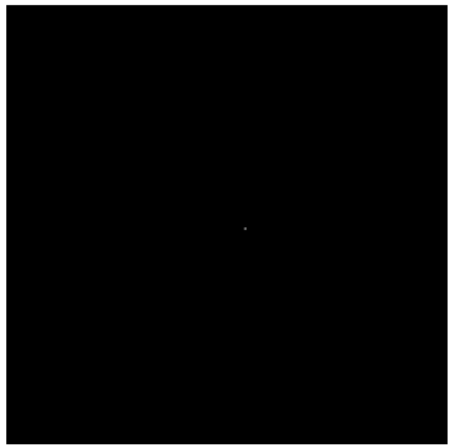


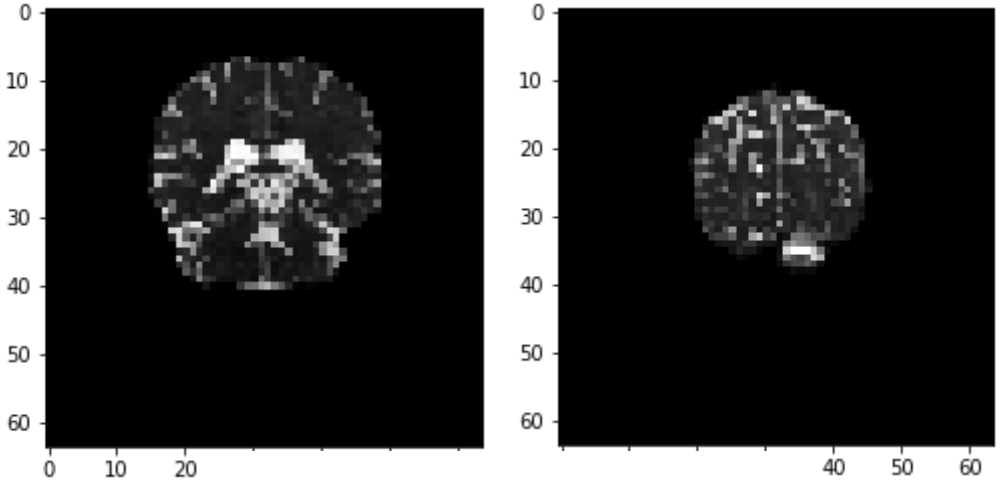
# Input Data

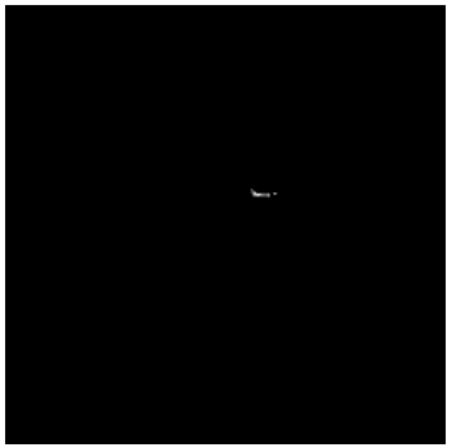
## T1wCE Picture



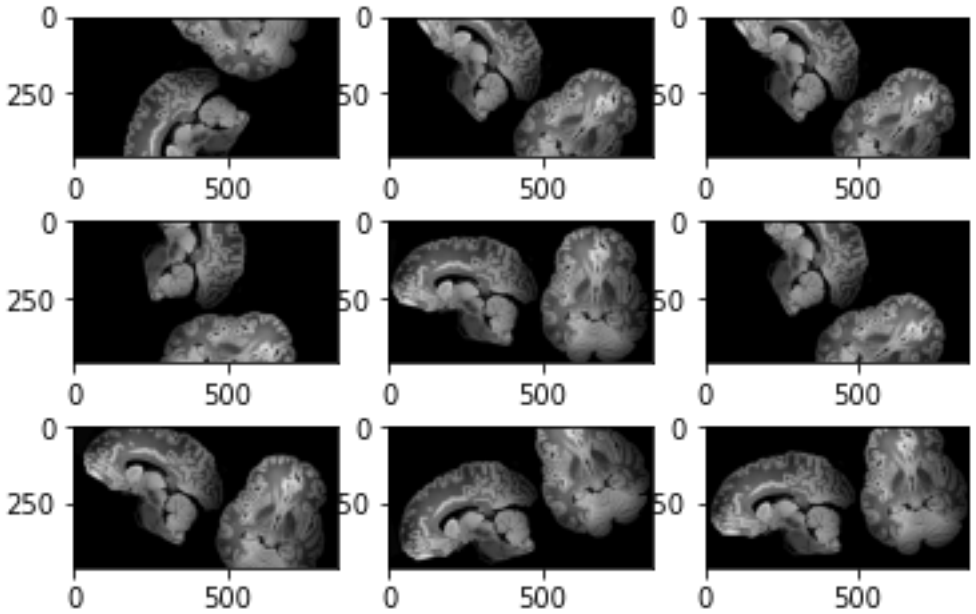
## T2w Picture



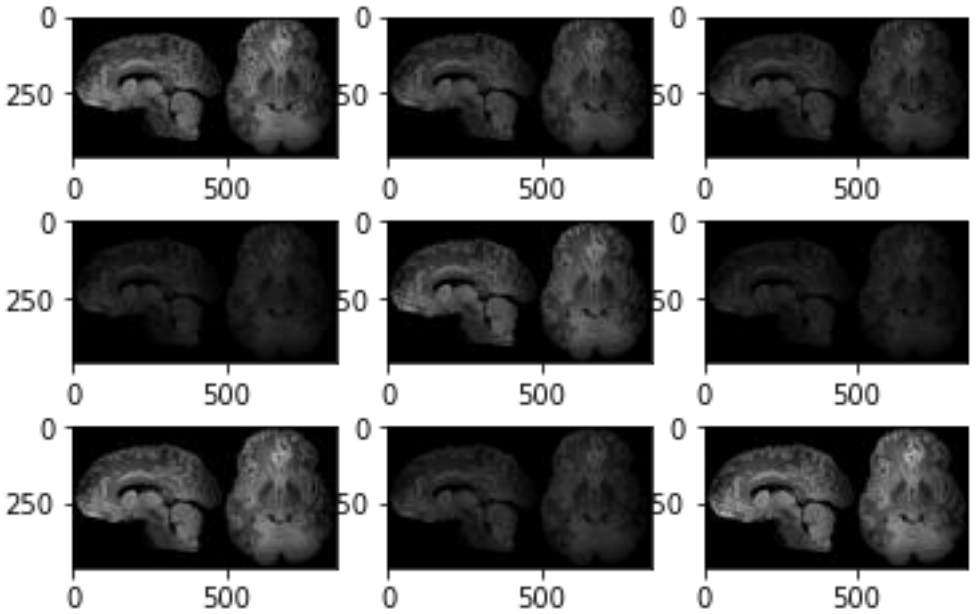



# Data Augmentation

## Rotation

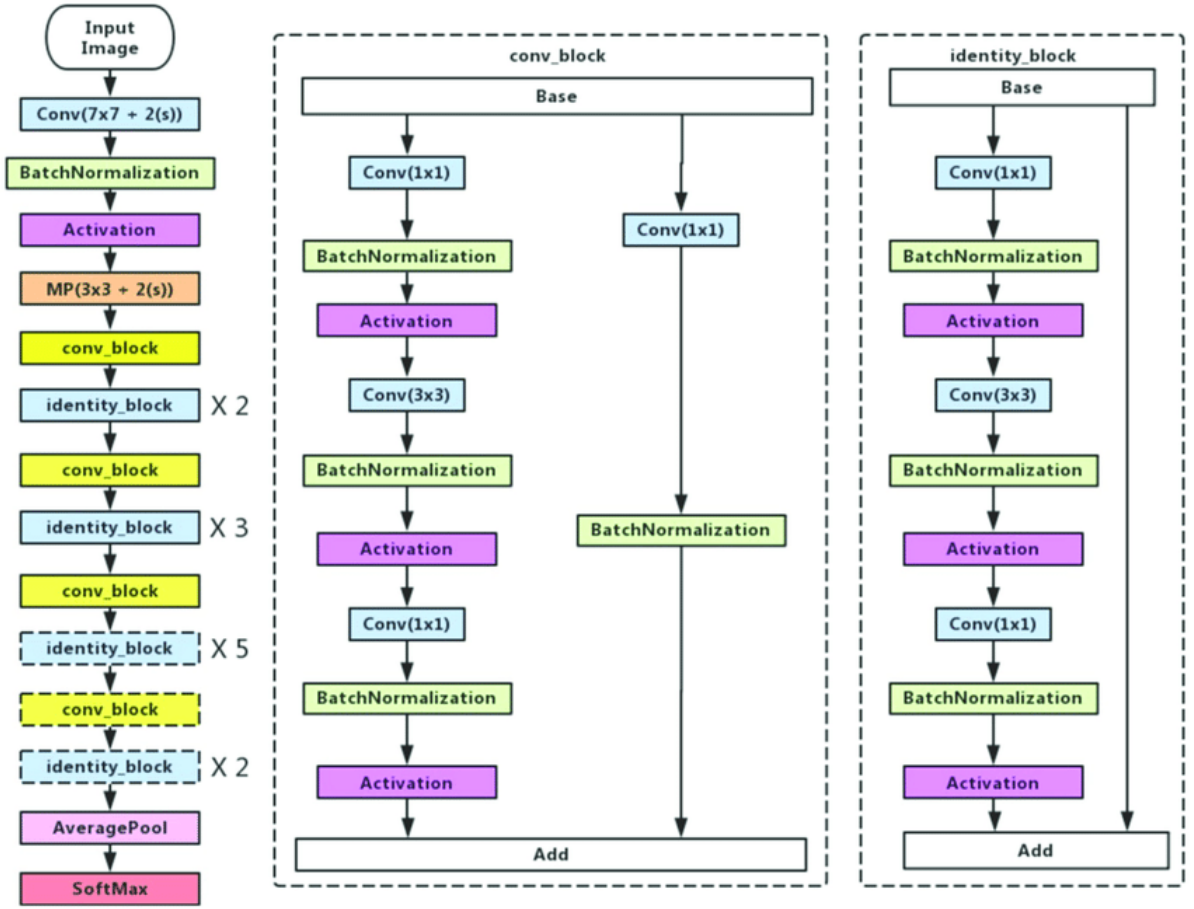



## Contrasting

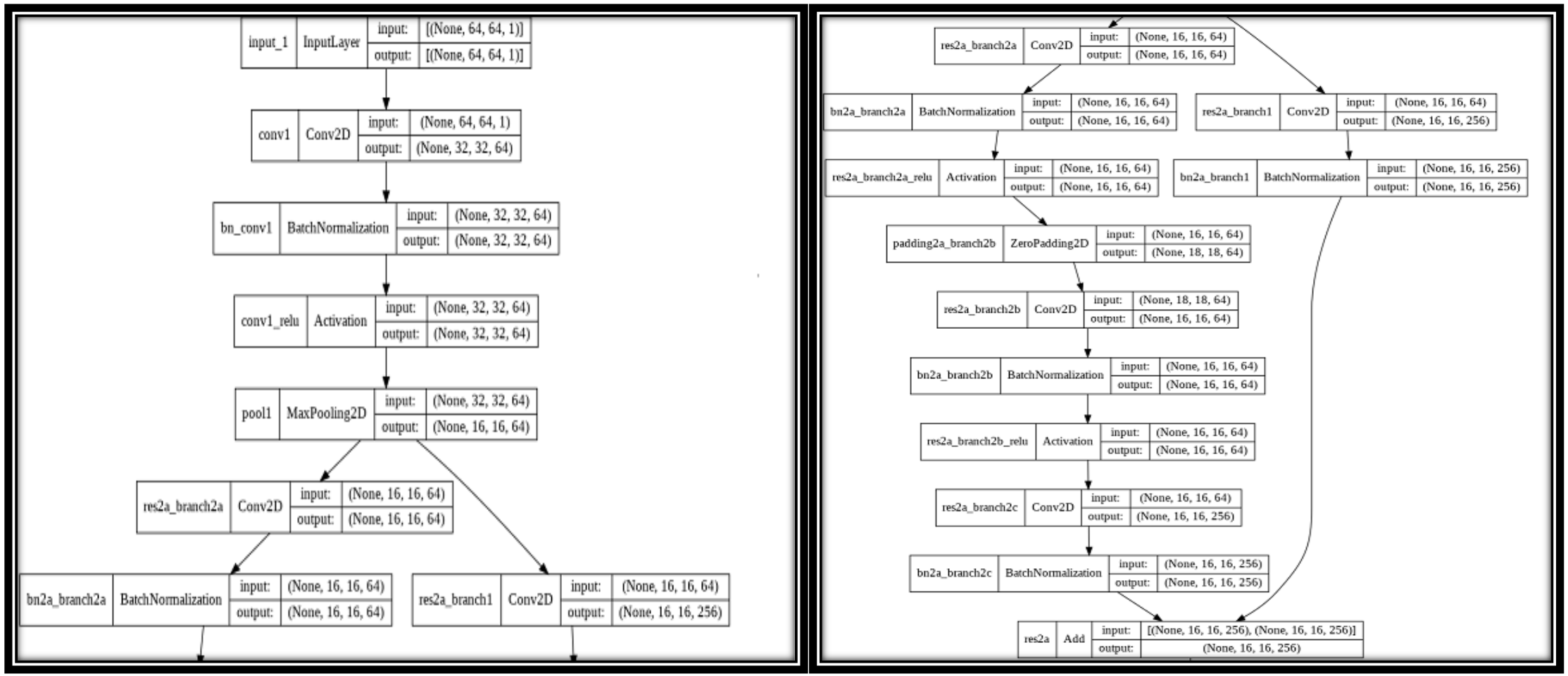


# Model 1:
# ResNet Model

# Resnet50

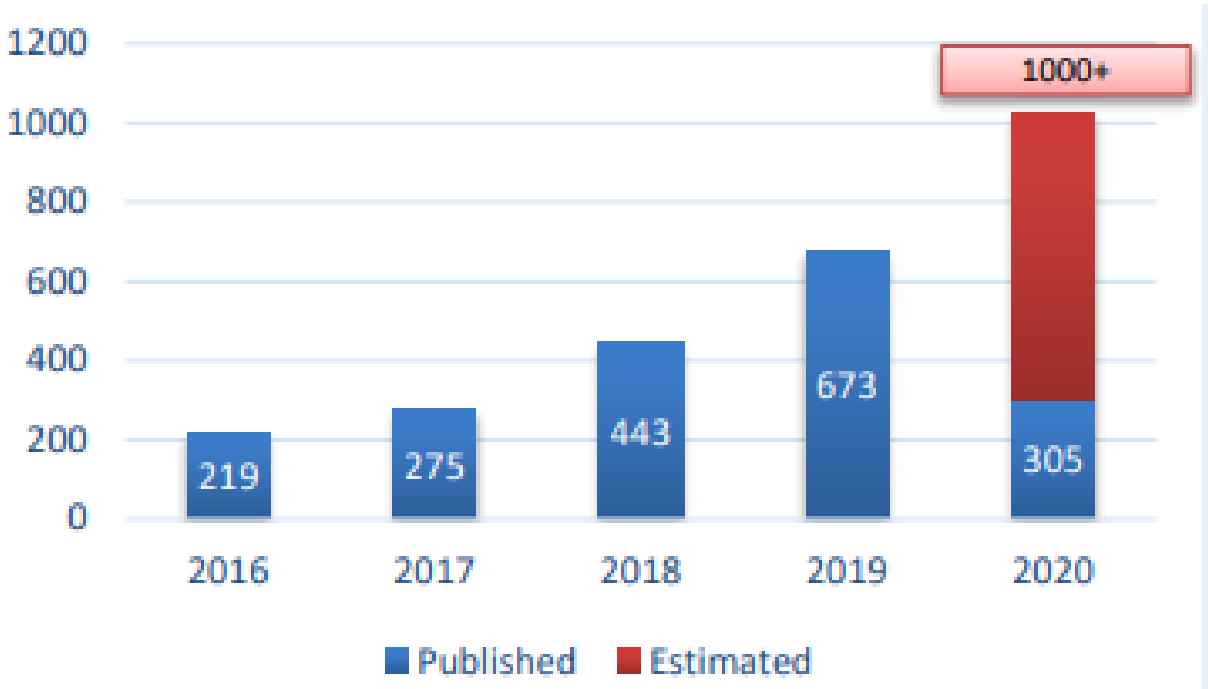

Input Image
Conv(7x7 + 2(s))
BatchNormalization
Activation
MP(3x3 + 2(s))
conv_block
identity_block
X 2
conv_block
identity_block
X 3
conv_block
identity_block
X 5
conv_block
identity_block
X 2
AveragePool
SoftMax
conv_block
Base
Conv(1x1)
BatchNormalization
Activation
Conv(3x3)
BatchNormalization
Activation
Conv(1x1)
BatchNormalization
Activation
Add
Conv(1x1)
BatchNormalization
identity_block
Base
Conv(1x1)
BatchNormalization
Activation
Conv(3x3)
BatchNormalization
Activation
Conv(1x1)
BatchNormalization
Activation
Add

# Our Model


input_1 InputLayer input: [(None, 64, 64, 1)] output: [(None, 64, 64, 1)]
conv1 Conv2D input: (None, 64, 64, 1) output: (None, 32, 32, 64)
bn_conv1 BatchNormalization input: (None, 32, 32, 64) output: (None, 32, 32, 64)
conv1_relu Activation input: (None, 32, 32, 64) output: (None, 32, 32, 64)
pool1 MaxPooling2D input: (None, 32, 32, 64) output: (None, 16, 16, 64)
res2a_branch2a Conv2D input: (None, 16, 16, 64) output: (None, 16, 16, 64)
bn2a_branch2a BatchNormalization input: (None, 16, 16, 64) output: (None, 16, 16, 64)
res2a_branch1 Conv2D input: (None, 16, 16, 64) output: (None, 16, 16, 256)
res2a_branch2a_relu Activation input: (None, 16, 16, 64) output: (None, 16, 16, 64)
padding2a_branch2b ZeroPadding2D input: (None, 16, 16, 64) output: (None, 18, 18, 64)
res2a_branch2b Conv2D input: (None, 18, 18, 64) output: (None, 16, 16, 64)
bn2a_branch2b BatchNormalization input: (None, 16, 16, 64) output: (None, 16, 16, 64)
res2a_branch2b_relu Activation input: (None, 16, 16, 64) output: (None, 16, 16, 64)
res2a_branch2c Conv2D input: (None, 16, 16, 64) output: (None, 16, 16, 256)
bn2a_branch2c BatchNormalization input: (None, 16, 16, 256) output: (None, 16, 16, 256)
bn2a_branch1 BatchNormalization input: (None, 16, 16, 256) output: (None, 16, 16, 256)
res2a Add input: [(None, 16, 16, 256), (None, 16, 16, 256)] output: (None, 16, 16, 256)

# Resnet50

**Number of CNN related publication in science direct**

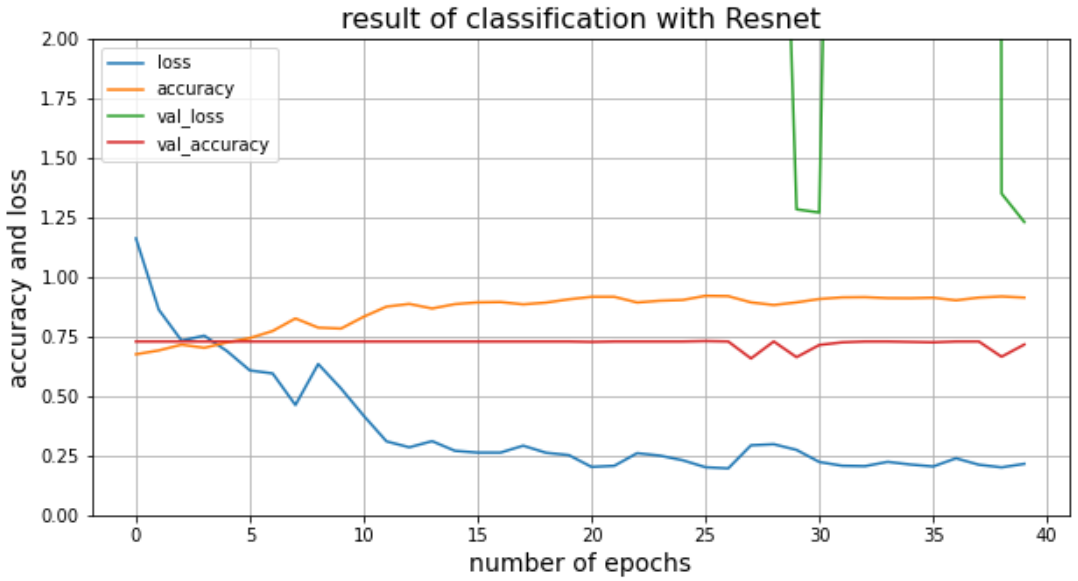



## Resnet50 Features

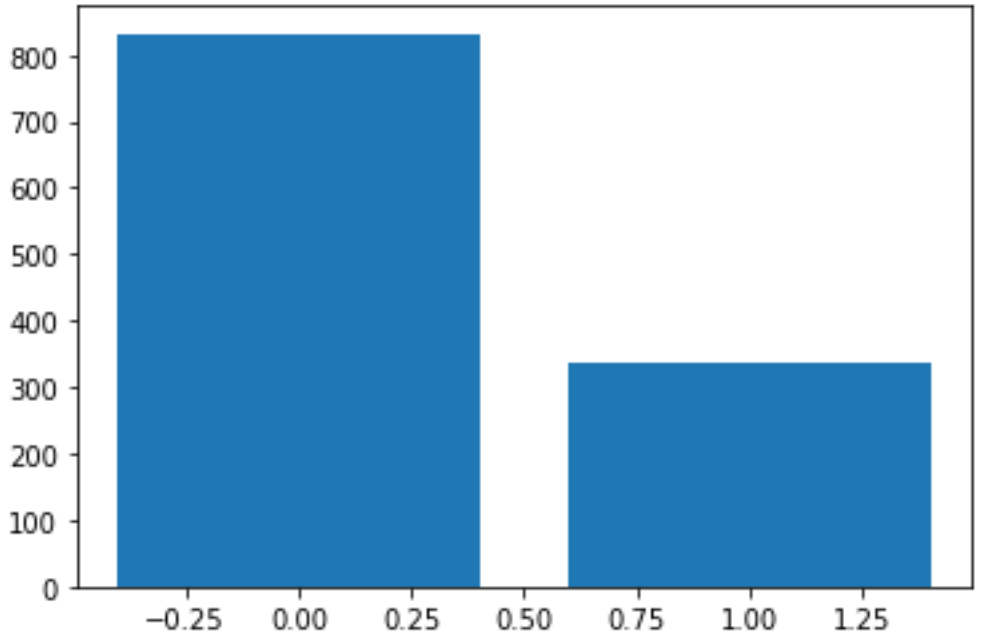
1. Faster computational time.
2. Higher achieved accuracy.
3. Better flow of optimization loss.
4. Compatible with transfer learning.
5. Most useful CNN algorithms from 2015 for classification.

# Result

## Accuracy and Loss




## Output prediction



# Discussion

- Based on the result on the training set and validation model got overfitted in the train set and needs more epochs for training.
- The result of the work is better than other famous CNN algorithms like VGG16 and VGG32.
- Achieved accuracy on test set is 74% which proves the needs for training the model with more procedure for regularization.
- The last layer which had been added is a dense layer with 2 neurons for classification specification about number of neurons can get changed based on number of classes

# Model 2:
# EfficientNet PytTorch 3D

GitHub open source:
https://github.com/tensorflow/tpu/tree/master/models/official/efficientnet

[1] Tan, M., & Le, Q.V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *ArXiv, abs/1905.11946*.

# Model Scaling

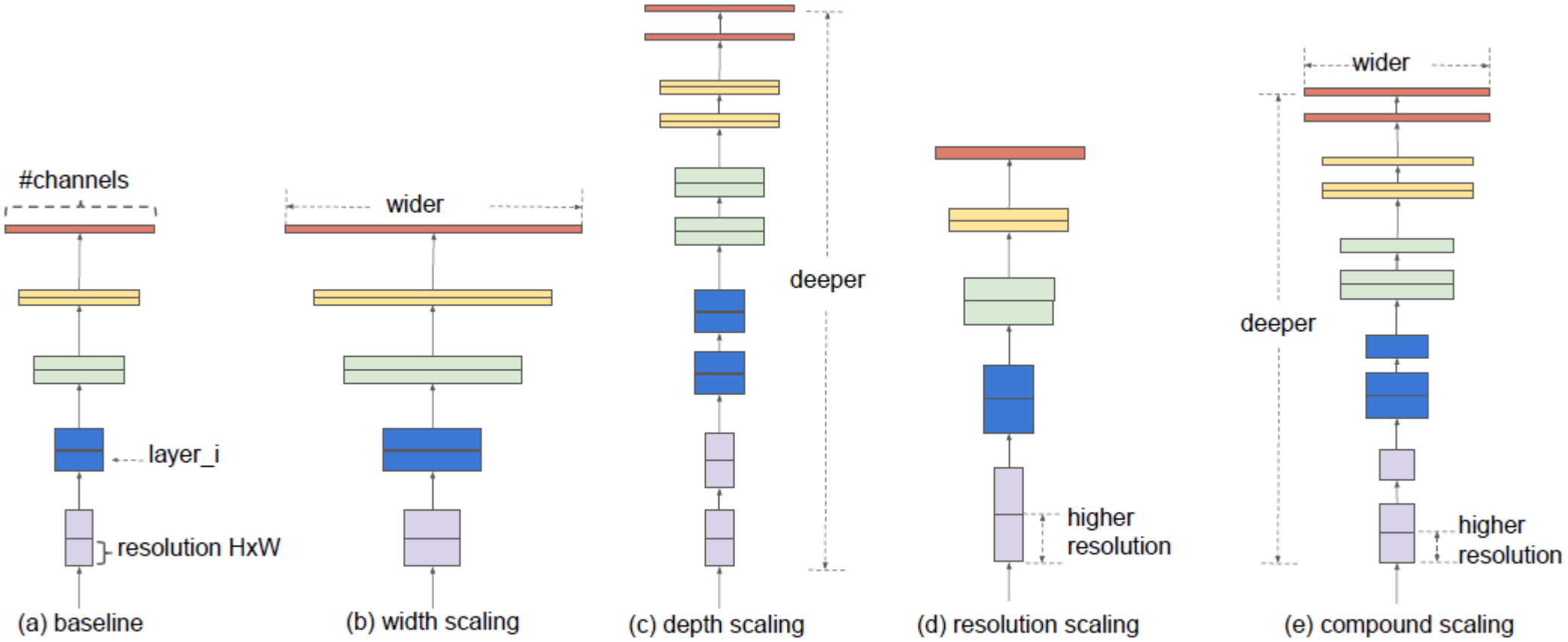
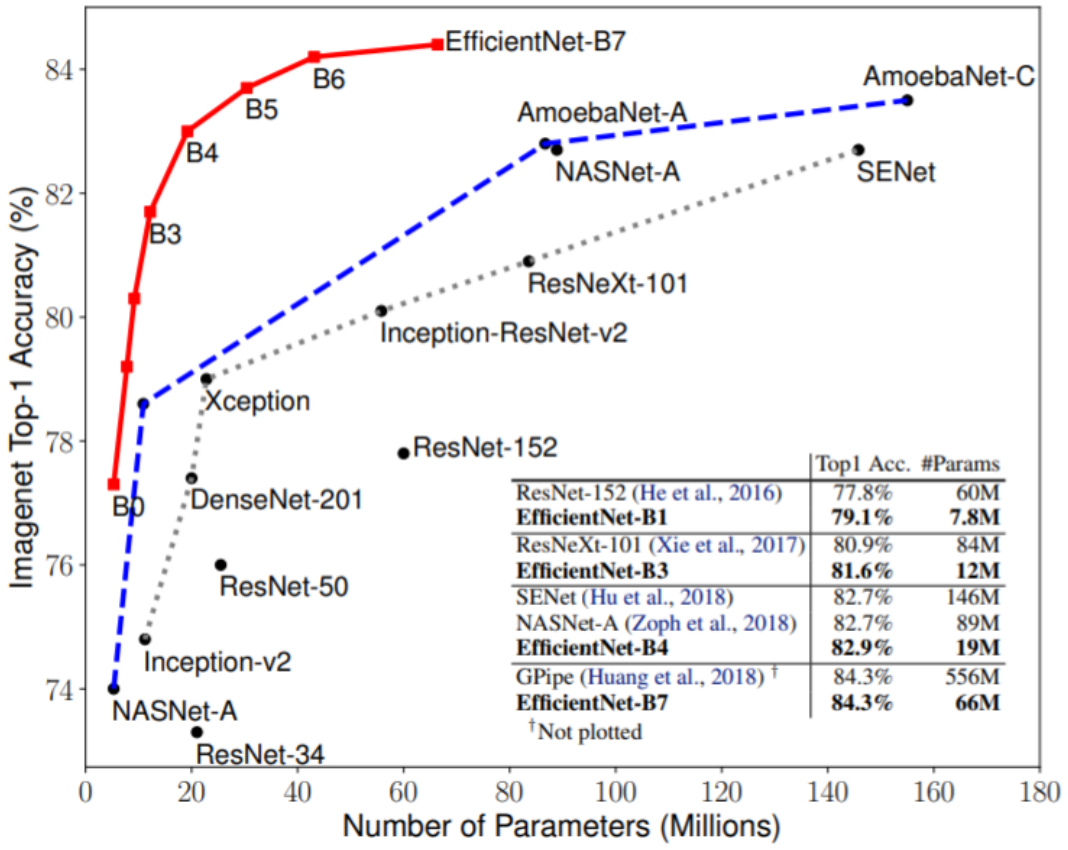


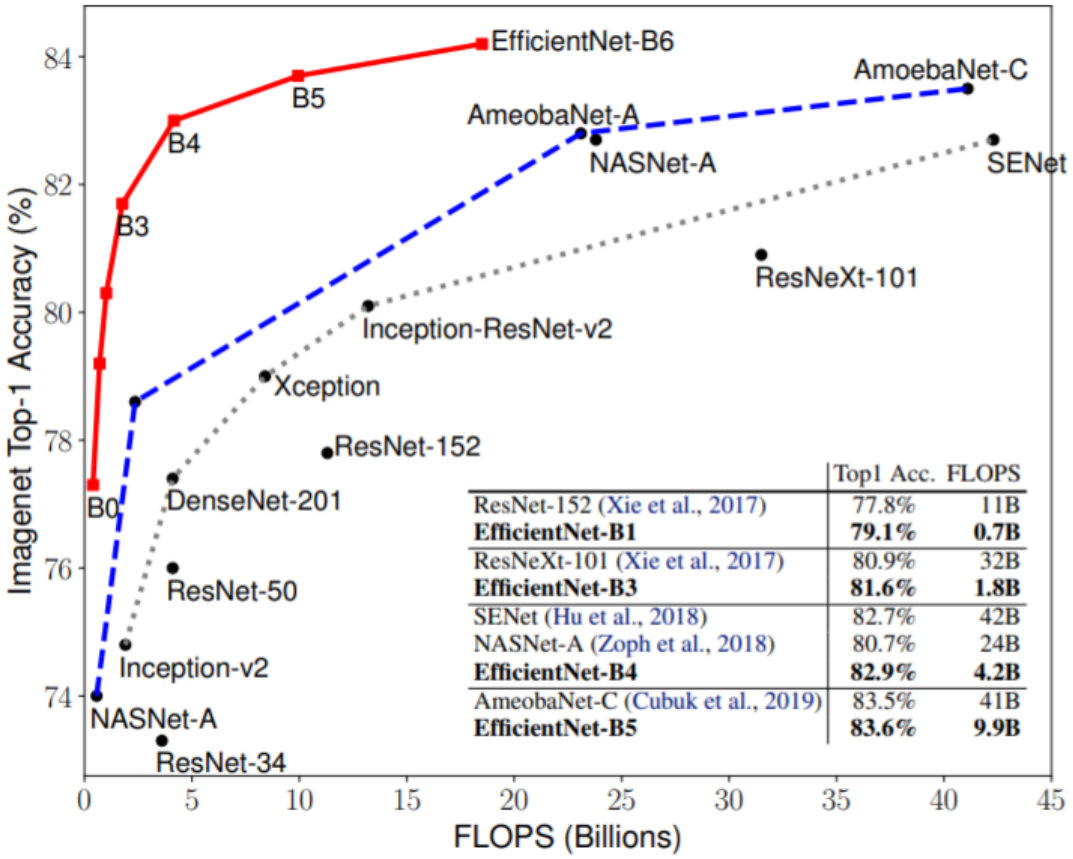

*Figure 2.* **Model Scaling.** (a) is a baseline network example; (b)-(d) are conventional scaling that only increases one dimension of network width, depth, or resolution. (e) is our proposed compound scaling method that uniformly scales all three dimensions with a fixed ratio.

# ImageNet Accuracy




| | Top1 Acc. | #Params |
|---|---|---|
| ResNet-152 (He et al., 2016) | 77.8% | 60M |
| **EfficientNet-B1** | **79.1%** | **7.8M** |
| ResNeXt-101 (Xie et al., 2017) | 80.9% | 84M |
| **EfficientNet-B3** | **81.6%** | **12M** |
| SENet (Hu et al., 2018) | 82.7% | 146M |
| NASNet-A (Zoph et al., 2018) | 82.7% | 89M |
| **EfficientNet-B4** | **82.9%** | **19M** |
| GPipe (Huang et al., 2018) † | 84.3% | 556M |
| **EfficientNet-B7** | **84.3%** | **66M** |

†Not plotted




| | Top1 Acc. | FLOPS |
|---|---|---|
| ResNet-152 (Xie et al., 2017) | 77.8% | 11B |
| **EfficientNet-B1** | **79.1%** | **0.7B** |
| ResNeXt-101 (Xie et al., 2017) | 80.9% | 32B |
| **EfficientNet-B3** | **81.6%** | **1.8B** |
| SENet (Hu et al., 2018) | 82.7% | 42B |
| NASNet-A (Zoph et al., 2018) | 80.7% | 24B |
| **EfficientNet-B4** | **82.9%** | **4.2B** |
| AmeobaNet-C (Cubuk et al., 2019) | 83.5% | 41B |
| **EfficientNet-B5** | **83.6%** | **9.9B** |

# Train

Use models with only one MRI type, then ensemble the 4 models

| | BraTS21ID | MGMT_value | MRI_Type |
|---|---|---|---|
| **446** | 645 | 0 | FLAIR |
| **529** | 777 | 1 | FLAIR |
| **420** | 607 | 1 | FLAIR |
| **406** | 589 | 0 | FLAIR |
| **475** | 690 | 1 | FLAIR |

| | BraTS21ID | MGMT_value | MRI_Type |
|---|---|---|---|
| **446** | 645 | 0 | T1w |
| **529** | 777 | 1 | T1w |
| **420** | 607 | 1 | T1w |
| **406** | 589 | 0 | T1w |
| **475** | 690 | 1 | T1w |

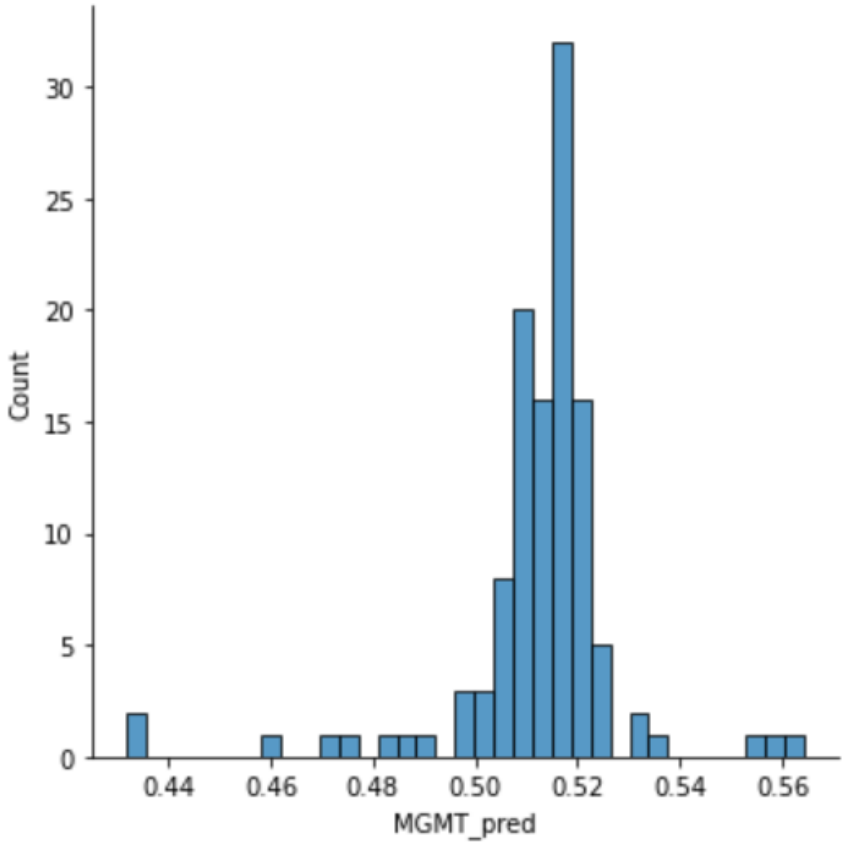
| | BraTS21ID | MGMT_value | MRI_Type |
|---|---|---|---|
| **446** | 645 | 0 | T1wCE |
| **529** | 777 | 1 | T1wCE |
| **420** | 607 | 1 | T1wCE |
| **406** | 589 | 0 | T1wCE |
| **475** | 690 | 1 | T1wCE |

| | BraTS21ID | MGMT_value | MRI_Type |
|---|---|---|---|
| **446** | 645 | 0 | T2w |
| **529** | 777 | 1 | T2w |
| **420** | 607 | 1 | T2w |
| **406** | 589 | 0 | T2w |
| **475** | 690 | 1 | T2w |

# Models & Test

```
Predict: FLAIR-e1-loss0.693-auc0.500.pth FLAIR (117, 3)
Predict: T1w-e8-loss0.683-auc0.529.pth T1w (117, 3)
Predict: T1wCE-e8-loss0.688-auc0.510.pth T1wCE (117, 3)
Predict: T2w-e5-loss0.693-auc0.533.pth T2w (117, 3)
Validation ensemble AUC: 0.5931
```

| | MGMT_value | MRI_Type |
|---|---|---|
| **BraTS21ID** | | |
| **1** | 0.511575 | T2w |
| **13** | 0.543372 | T2w |
| **15** | 0.516285 | T2w |
| **27** | 0.536630 | T2w |
| **37** | 0.526855 | T2w |
| **...** | ... | ... |
| **826** | 0.517826 | T2w |
| **829** | 0.513860 | T2w |
| **833** | 0.518291 | T2w |
| **997** | 0.498467 | T2w |
| **1006** | 0.516734 | T2w |



# Next:

Compare and choose the best model;
Make products easier to applicate.

# THANK YOU