Transfer Learning approach for Brain Tumor Classification on Imbalanced MRI dataset

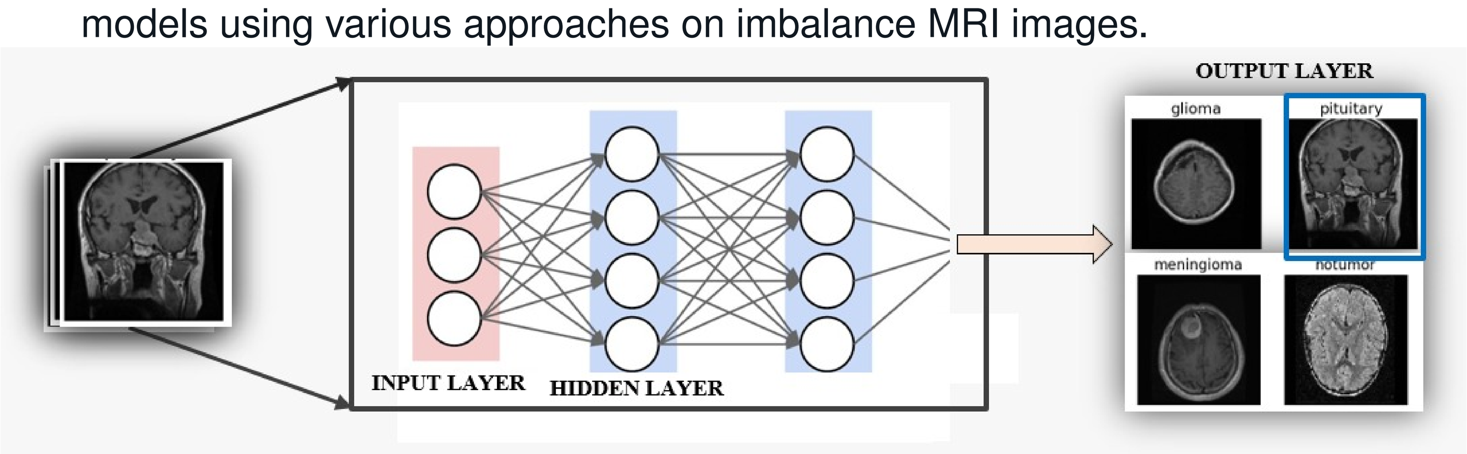
Raza Imam, Mohammed Talha Alam, Mohammed El-Amine Azz

Mohamed bin Zayed University of Artificial Intelligence

MOHAMED BIN ZAYED UNIVERSITY OF ARTIFICIAL INTELLIGENCE

Introduction

- Deep learning and Transfer learning models have been proposed with higher accuracies in recent literatures for classifying Brain Tumor MRI images. However most of such proposals are focused on balanced data.
- Hence, approaches to account for the imbalance in data, as well as focusing on the precise classification of brain cancer in real world scenarios, are needed.
- We introduce a 'Transfer Learning + CNN model' rather of just 'Transfer Learning model with Fine Tuning' and compared 8 of such Transfer Learning models using various approaches on imbalance MRI images.

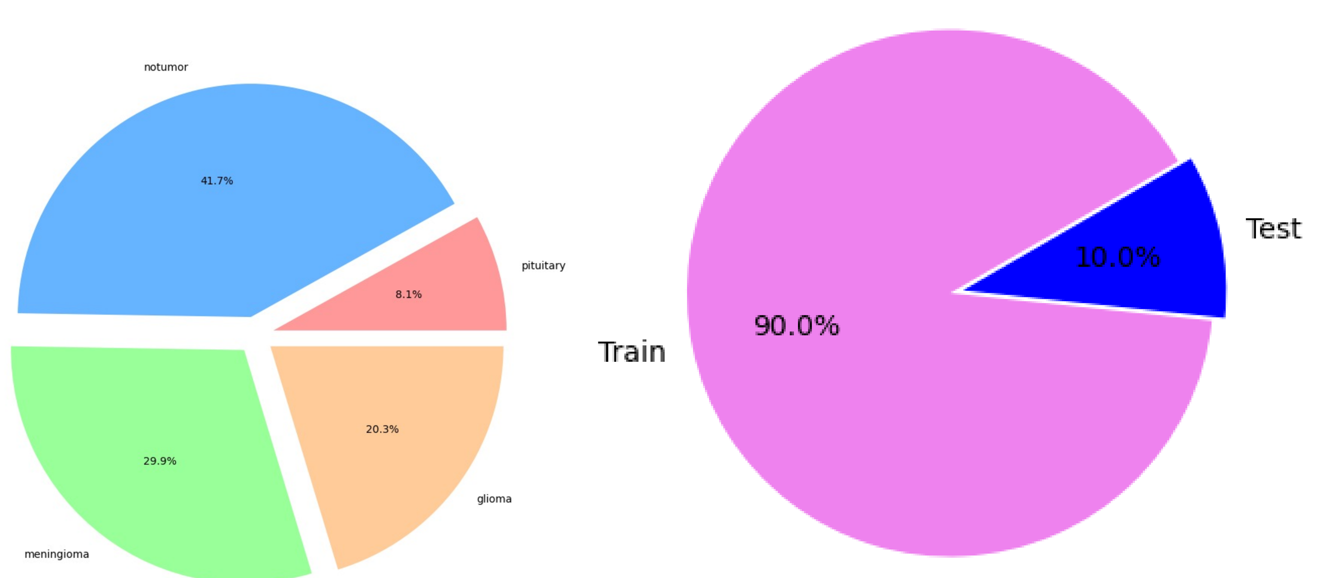


Research Objectives

- We employ 8 transfer learning models integrated with CNNs to increase the classification accuracy on different types of brain cancer (no tumor, glioma, meningioma, and pituitary cancer).
- We experiment with 5 different approaches to deal with imbalanced datasets such as- **Changing loss functions**: 1.Focal loss 2.Cross Entropy, and **Oversampling methods**: 3.Data Augmentation, 4.SMOTE 5.ADASYN.
- Our empirical evaluation of the models under the different approaches are assessed on Metrics including Accuracy, Precision, Recall, and F1 Score.

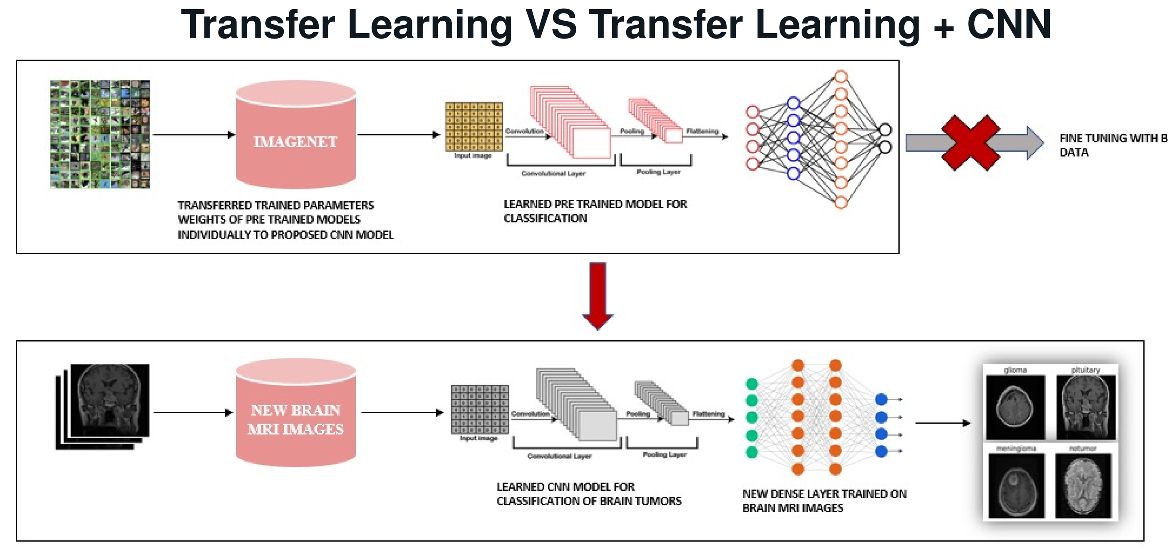
Dataset

- The accumulated imbalance dataset contains about 4200 MRI images with 4 classes: no tumor (1760), glioma (858), meningioma (1265) and pituitary (341) cancer.
- For validation, the dataset is divided into train and test with 90:10 ratio.
- The Testing dataset is used to evaluate all of the accuracy claims made about the models in the Results and Comparisons section.



Methods

Transfer Learning VS Transfer Learning + CNN



Loss Functions

- **Focal Loss** Lower class counts have higher weights in errors compared to cross-entropy, i.e. downweigh the easy examples and focus on hard examples.

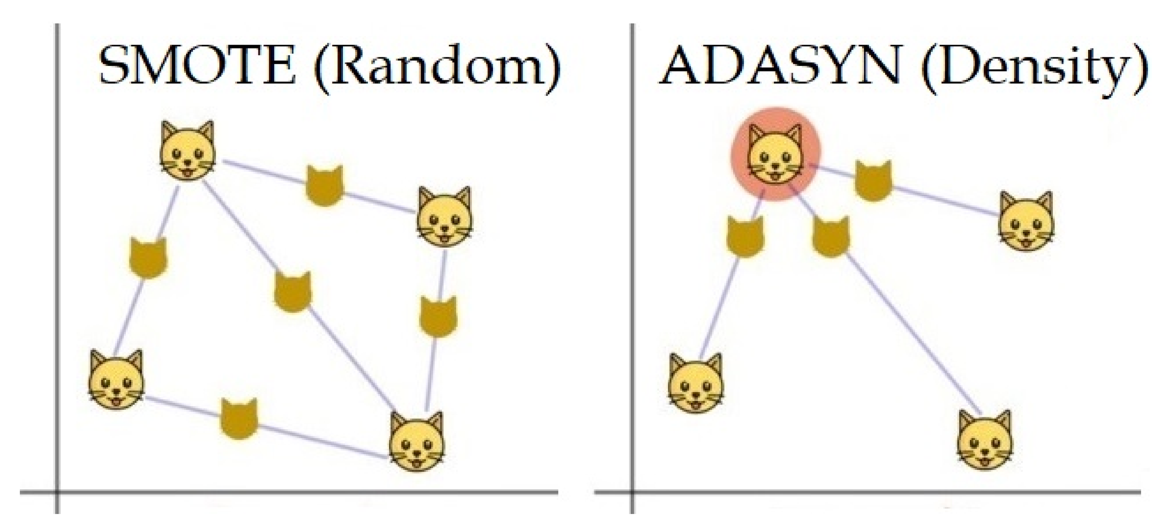
$$FL = -(1 - P_t)^y log(P_t)$$

- **Cross Entropy** Loss increases as the predicted probability diverges from the actual label.

$$CE = -(ylog(p) + (1 - y)log(1 - p))$$

Balancing Methods

- **Augmentation** Zooming, Cropping, Sharpness alteration
- **SMOTE** Applies KNN approach where it selects K nearest neighbors, joins them and creates the synthetic samples in the space.
- **ADASYN** Uses a density distribution to decide the number of synthetic samples that must be generated for each minority sample by adaptively changing the weights of the different minority samples to compensate for the skewed distributions.

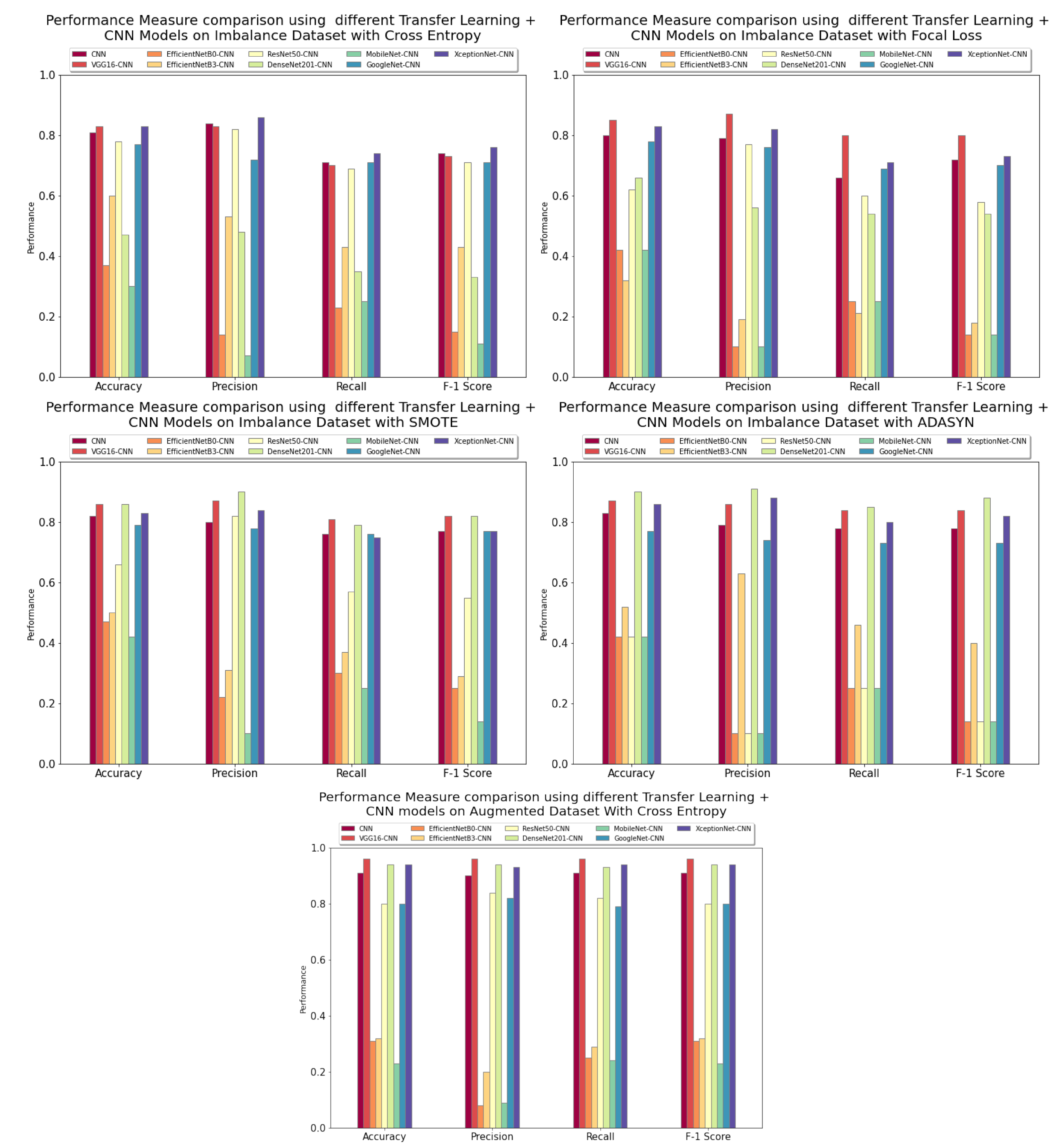


Results & Comparisons

Table. Accuracy Comparison on Unbalanced Dataset

Model	CE	Focal Loss	Augment CE	SMOTE CE	ADASYN CE
CNN	0.8125	0.80	0.91	0.82	0.83
VGG16-CNN	0.83	0.85	0.96	0.86	0.87
EfficientNetB0-CNN	0.37	0.42	0.31	0.47	0.42
EfficientNetB3-CNN	0.60	0.32	0.32	0.50	0.52
ResNet50-CNN	0.78	0.62	0.80	0.66	0.42
DenseNet201-CNN	0.47	0.66	0.94	0.86	0.90
MobileNet-CNN	0.30	0.42	0.23	0.42	0.42
GoogleNet-CNN	0.77	0.78	0.80	0.79	0.77
XceptionNet-CNN	0.83	0.81	0.94	0.83	0.86

- We have incorporated 8 Transfer learning + CNN models and compared them on 5 methodologies including Augmentation, Focal Loss, SMOTE, and ADASYN.
- In terms of accuracy, VGG16 performed the best among all 5 approaches overall, while MobileNet and EfficientNets are among the worst performers.
- In terms of loss functions on Imbalanced Data, Focal loss was better for Models including VGG16, EffNetB0, DenseNet201, MobileNet, and GoogleNet, while the other models performed better with cross entropy.
- In terms of balancing methods, Data Augmentation achieved the best overall accuracy among the 3 approaches.



- In terms of Average Precision, Recall, and F1 Scores, VGG16 with Data Augmentation performed the best among every combination of approaches.

Conclusion & Future Works

- Using different approaches and evaluation metrics, VGG16 Transfer Learning with CNN method performed the best overall.
- Improve the accuracy of VGG16 and DenseNet models via Hyperparameter tuning and extensive GridSearchCV.
- Focusing on increasing the performance of lightweight models like MobileNet as it will be more applicable in real-world scenarios.

References