CLASSIFICATION CHALLENGE :

CADx PROJECT: SKIN LESION CLASSIFICATION CHALLENGE USING DEEP LEARNING (DL) APPROACH

Presentation by:

Daniel Tweneboah Anyimadu & Taofik Ahmed Suleiman

Supervised by:

Prof. Xavier Llado & Prof. Arnau Oliver

TABLE OF CONTENTS

1 ABSTRACT

- This study focuses on the classification of skin lesions through a deep-learning (DL) approach employing a range of both pre-trained DL models and customized CNNs.

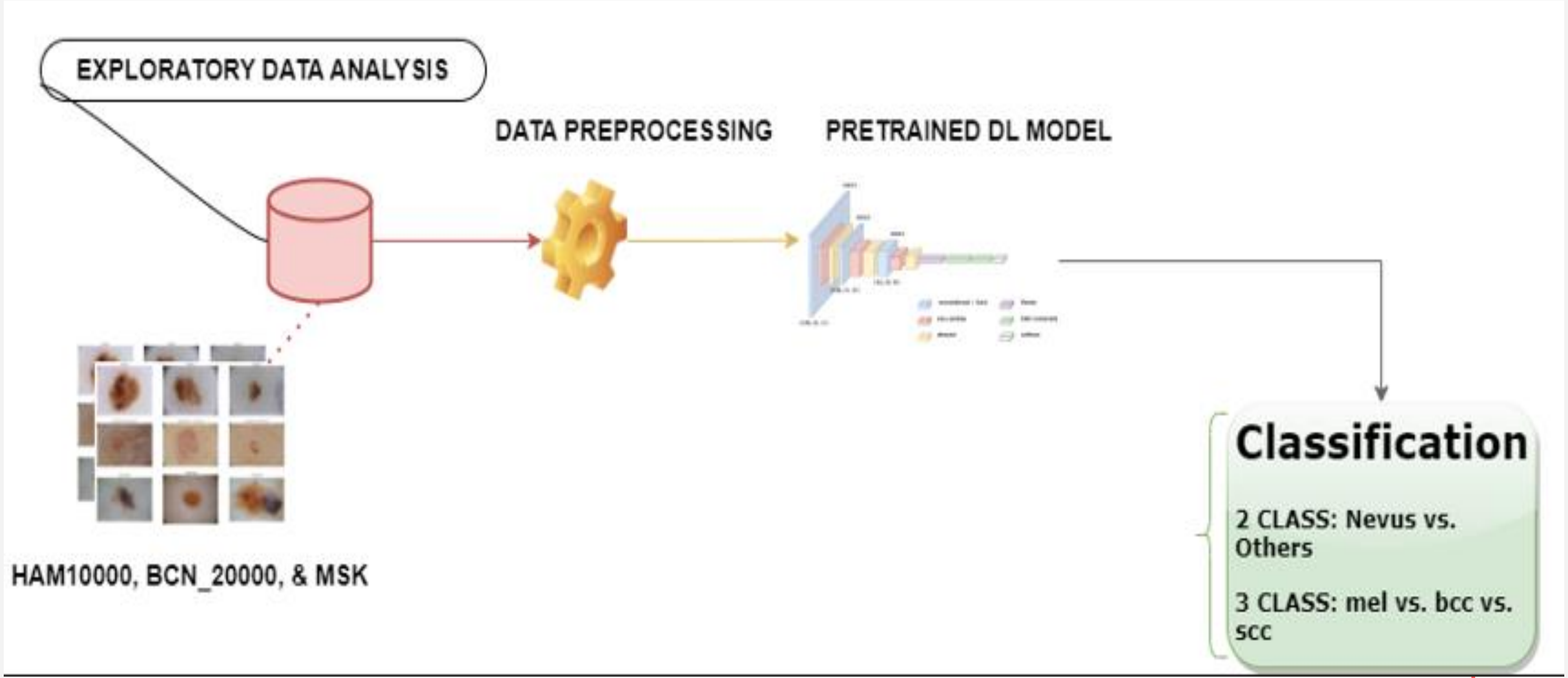
- Our approach addresses both two-class and three-class problems, improving classification accuracy despite huge class imbalances and lesion variations.

- The challenge dataset was provided by our course organizers and includes images from the HAM10000, BCN_20000, and MSK datasets, offering a rich variety of dermatological samples for analysis. The project involves exploratory data analysis, data preprocessing, feature extraction using DL models, model training, classification, and performance evaluation.

- Two-class problems are assessed using overall accuracy, while three-class problems are evaluated using kappa scores on a test dataset.

- The ensemble model (MobileNet + InceptionResNetV2) achieved an accuracy of 0.90 on the two-class problem where as (MobileNet + InceptionResNetV2 + ResNet50)achieved a kappa value of 0.86 on the 3-class.

GRAPHICAL ABSTRACT



2

EXPLORATORY DATA ANALYSIS

i

Data Visualization

Exploratory Data Analysis (EDA)

- 2 Class Distribution Plot :

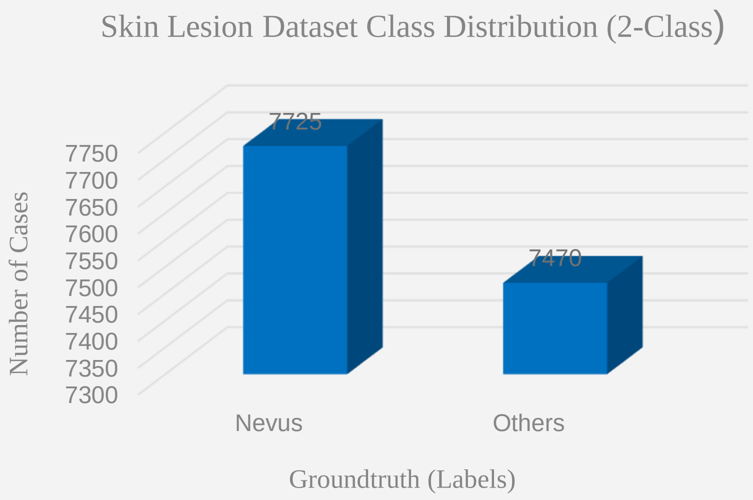


Fig 1: 2 class distribution plot

- 3 Class Distribution Plot :

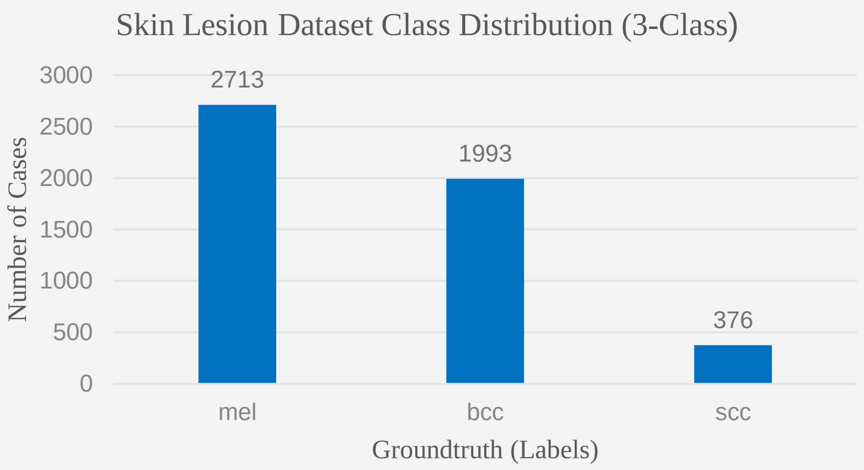


Fig 2: 3 class distribution plot

- A visualization of class imbalance shown in the 3-class distribution plot.

3 DATA PRE-PROCESSING

- i Vignette Frame Removal
- ii Stratified Shuffle Split
- iii Data Augmentation
- iv Resampling Technique
- v Computing Class Weight

Data Preprocessing: Vignette Frame Removal – 2 Class

❑ Vignette Frame Removal:

- Define a function that takes an image and a threshold as input parameters. It then detects the threshold where darkening occurs and crops the image accordingly.

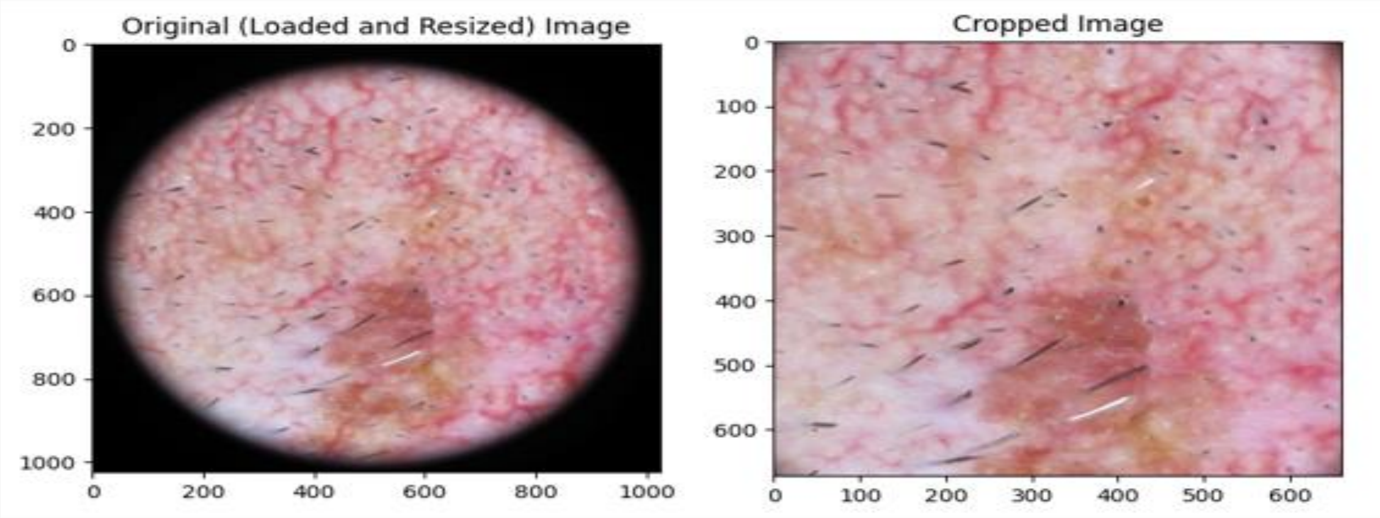


Fig 3: Vignette frame removed image

Data Preprocessing: StratifiedShuffleSplit and Data Augmentation

❑ Stratified Shuffle Split:

- Maintains class distribution in training and fake_test datasets.
- E.g. 7725 (O) = 6180 (T) + 1545 (f_t)

❑ Data Augmentation:

- Increases diversity in training dataset by applying random transformations including rescaling, zooming, horizontal and vertical flips.

Fig 4: Visualization of augmented images

Data Preprocessing: Handling Imbalanced Data – 3 Class

❑ Upsampling and Downsampling:

- Resampling techniques were strategically applied to achieve a more realistic representation of data instances, ensuring a balanced and reflective dataset for improved model training.

❑ Compute Class Weights:

- Appropriate weights are assigned to each class to account for imbalances during model training. Leveraged "compute_class_weight" function with "balanced" option.

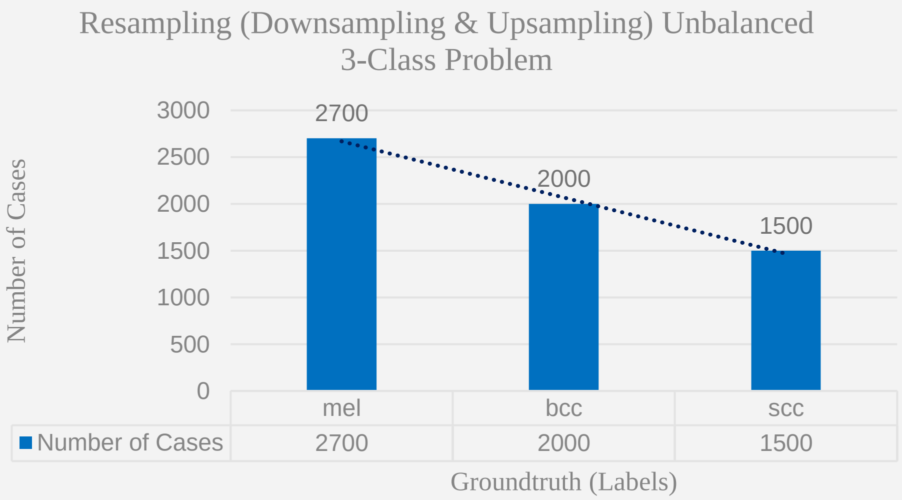


Fig 5: Resampled data visualization

4 DEEP LEARNING MODEL

- i. Pre-trained Models
- ii. Custom Layers Built
- iii. Best Models
- iv. Callbacks

Deep Learning (Model Training)

❑ Deep Learning Models:

- A subfield of ML that focuses on the development and applications of artificial neural networks, specifically deep neural networks, to model and understand complex patterns and relationships in data.

❑ EfficientNet V2 B0 model:

- It introduces a compound scaling method to uniformly scale all dimensions of depth, width, and resolution.

❑ Vision Transformer:

- It breaks down an image into fixed-size patches, linearly embeds them, and processes them using transformer blocks.

❑ DenseNet121/210:

- It connects each layer to every other layer in a feed-forward fashion.

❑ ResNet50 :

- It introduces skip connections to jump over some layers, addressing the vanishing gradient problem.

Deep Learning (Model Training)

- **MobileNet:**
  - It utilizes depth-wise separable convolutions to reduce the number of parameters and computational cost.

- **Swin Transformer:**
  - Uses self-attention mechanisms to capture long-range dependencies in images.

- **RegNet:**
  - It focuses on providing a scalable architecture by tuning hyperparameters systematically.

- **InceptionResNetV2:**
  - An extension of Google's Inception architecture with residual connections.

- **VGG16:**
  - It has 16 weight layers and is characterized by its use of small receptive filters (3x3).

Deep Learning (Best Models)

- **InceptionResNetV2, ResNet50, MobileNet121**

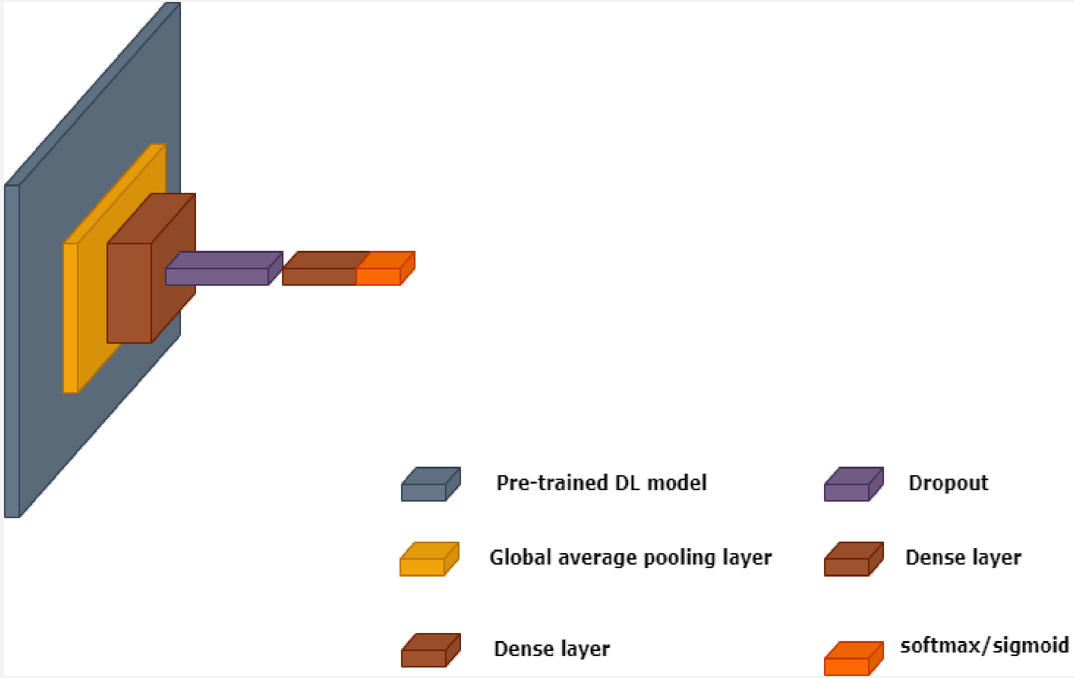


Fig 6: Custom layers built on top of the pre-trained models

Our custom model, built on top of the pre-trained models, include the following layers:

- Feature extractor
- Reduces spatial dimensions to capture features of interest
- Learns complex patterns from extracted features
- To prevent overfitting
- For binary/multiclass classification

Inception-ResNet-V2

Custom Layer on top of inceptionresnetv2:

- Pre-trained InceptionResNetV2
- Global Average Pooling layer
- Dense Layer with 128 units & ReLu Activation
- Dropout Layer
- Dense layer with sigmoid/softmax activation

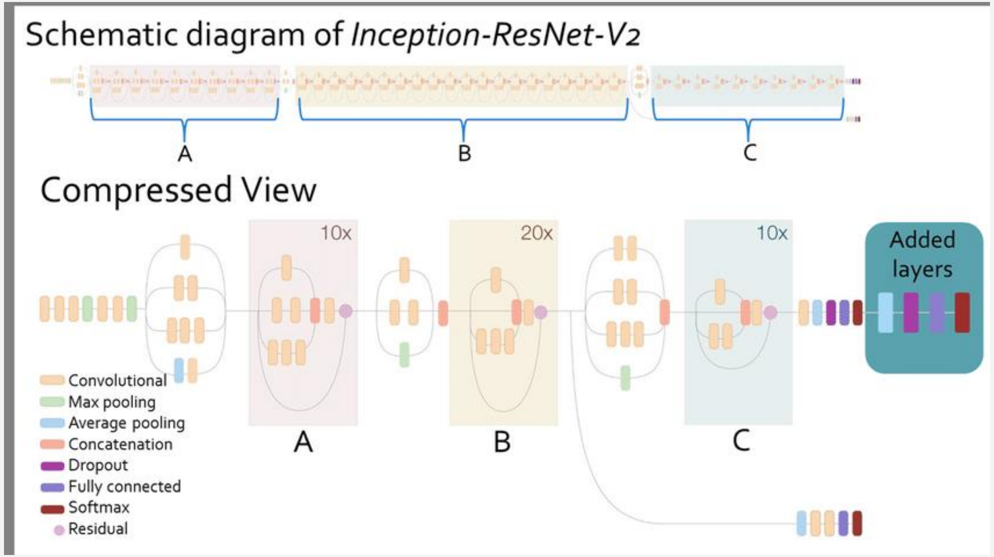


Fig 7: InceptionResNet Architecture

ResNet50

Custom Layer on top of resnet50:

- ❑ Pre-trained ResNet50
- ❑ Global Average Pooling layer
- ❑ Dense Layer with 256 units & ReLu Activation
- ❑ Dropout Layer
- ❑ Dense layer with sigmoid/softmax activation

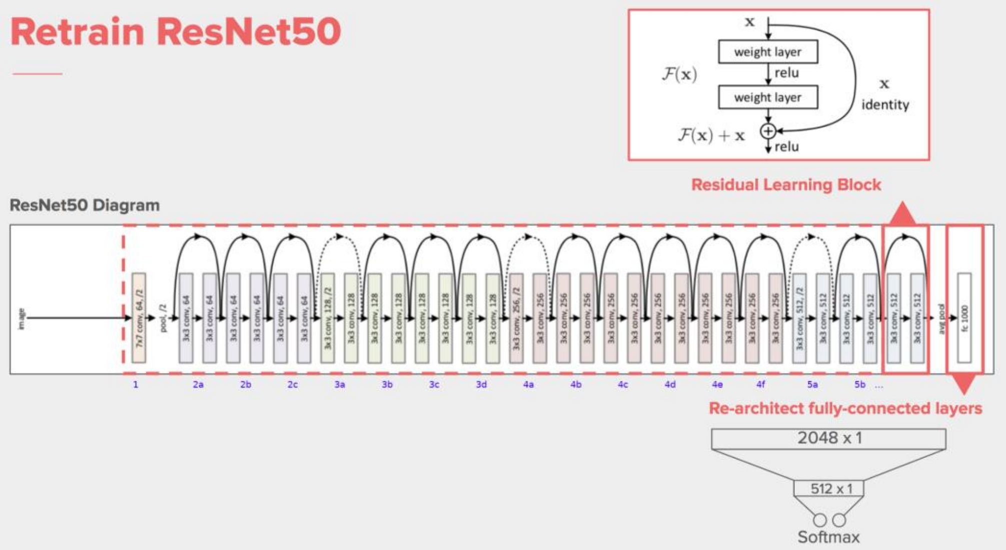


Fig 8: ResNet50 Architecture

MobileNet

Custom Layer on top of mobilenet:

- ❑ Pre-trained MobileNet [-6]
- ❑ Flatten layer
- ❑ Dense Layer with 1024 units & ReLu Activation
- ❑ Dropout Layer
- ❑ Dense layer with sigmoid/softmax activation

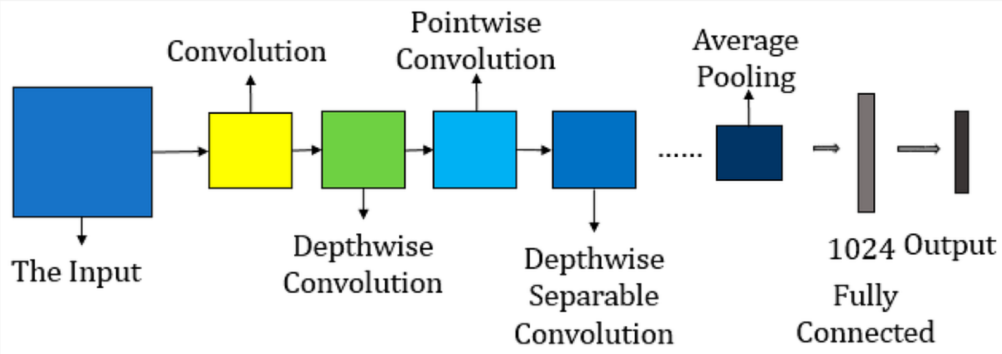


Fig 9: MobileNet Architecture

Callbacks

- Contribute to the stability, efficiency, and generalization of the trained models by dynamically adjusting learning rates, preventing overfitting, and saving the best models during training.

- ReduceLROnPlateau:

- It is used to adjust the learning rate during training based on a specified metric.

- EarlyStopping:

- Used to stop training when a monitored metric has stopped improving, avoiding overfitting.

- ModelCheckpoint:

- Saves the model after every epoch if the specified metric (monitor) improves.

5

PERFORMANCE EVALUATION

i 2-CLASS ACCURACY

ii 3-CLASS KAPPA

PERFORMANCE EVALUATION (ACCURACY) – 2 CLASS

- Table 1: TWO-CLASS PROBLEM (Batch Size = 32)

MODEL (DL)	# OF EPOCHS (TRAINING)	ACCURACY
EfficientNet V2 B0 model	10, 30	0.8406, 0.8527
Vision Transformer	10	0.8274
DenseNet121	30	0.8248
ResNet50	30	**0.8622**
InceptionResNetV2	30	**0.8728**
VGG16	30	0.8111
MobileNet	20	**0.8604**
ResNet50 + InceptionResNetV2 + MobileNet		**0.88 (+w)**

PERFORMANCE EVALUATION (ACCURACY) – 2 CLASS

- Table 2: TWO-CLASS PROBLEM (Batch Size = 64)

MODEL (DL)	# OF EPOCHS (TRAINING)	ACCURACY
ResNet50	100	0.8704
InceptionResNetV2	100, 150	0.8999, **0.8999**
MobileNet	100, 150	0.76, 0.7856
ResNet50 + MobileNet + InceptionResNetV2		**0.89**
ResNet50 + InceptionResNetV2		**0.90**
MobileNet + InceptionResNetV2	100, 150	**0.90, 0.90**

PERFORMANCE EVALUATION (KAPPA) – 3 CLASS

- Table 3: THREE-CLASS PROBLEM (mel: 2700, bcc: 2000, scc: 1500)

MODEL (DL)	**# OF EPOCHS (TRAINING)**	**KAPPA**
DenseNet121	20	0.6225
DenseNet210	30	0.6902
MobileNet	30, 50	**0.7873,** 0.7802
ResNet50	30, 50, 100	0.7575, **0.7906**, 0.7623
InceptionResNetV2	30, 50, 100	**0.8446,** 0.82**,** 0.8301
EfficientNet V2 B0	30	0.7030
MobileNet + InceptionResNetV2		0.8506 (+p), 0.8518 (+w)
MobileNet + InceptionResNetV2 + ResNet50		0.8534 (+w)**, 0.8577 (+w)**

EXPLORATION ON MONAI FRAMEWORK

Medical Open Network for AI (MONAI) is an open-source PyTorch-based framework for healthcare imaging, fostering collaboration among researchers. It offers end-to-end training workflows, flexible pre-processing, compositional APIs, and domain-specific implementations, ensuring standardized and optimized deep learning model creation and evaluation in medical imaging.

- ❑ Table 4: TWO-CLASS PROBLEM

MODEL (DL)	# OF EPOCHS (TRAINING)	ACCURACY
Monai - DenseNet121	10	0.7582
Monai - DenseNet169	10, 100	0.7882, 0.7966
Monai – DenseNet201	10	0.7690
Monai – DenseNet264	10	0.7861

CONCLUSION

- Based on the results, the combination of preprocessing techniques including Stratified Shuffle Split, Data Augmentation, and introducing Class Weights as well as the use of both MobileNet and InceptionResNetV2 classifiers as ensemble model enhanced the accuracy performance of our binary classification model.

- Additionally, leveraging the same methodology + resampling techniques with the ensemble of MobileNet + InceptionResNetV2 + ResNet50 classifier led to improved kappa performance in our multi-classification model.

- **THANK YOU FOR YOUR ATTENTION!!!**