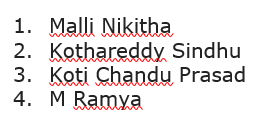
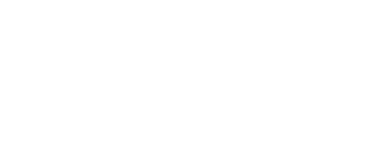
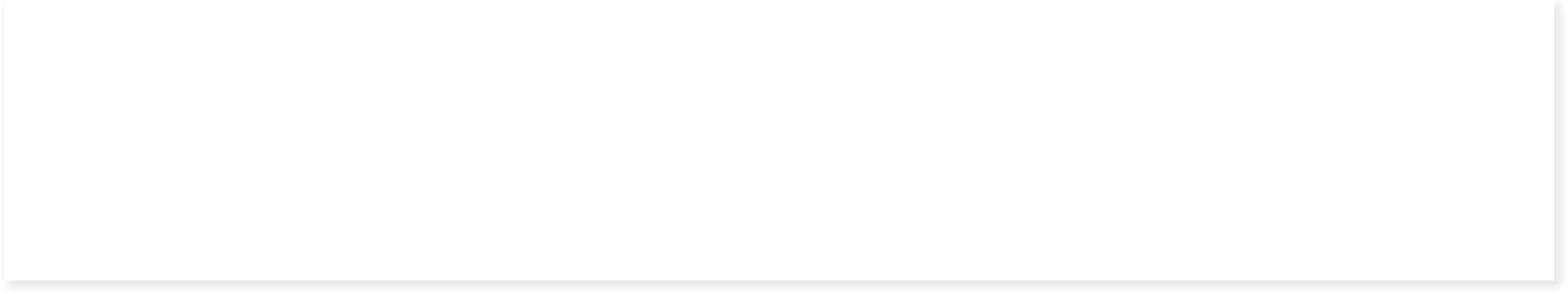
Foundations of Deep

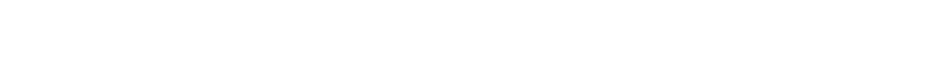
Learning Project



* Loading and Sampling
* EDA: Exploratory Data Analysis
* Preprocessing + Splitting data into Training and Validation Set
* Classification: Neural Network from scratch and Transfer Learning



•



* Selection and Comparison of the Best Models
* Fitting on Test data and Best Model Choice

Rice Image Dataset 

* [https://www.kaggle.com/datasets/muratkokludataset/rice-image-dataset ●](https://www.kaggle.com/datasets/muratkokludataset/rice-image-dataset)75K images.
* 5 types of rice grains:



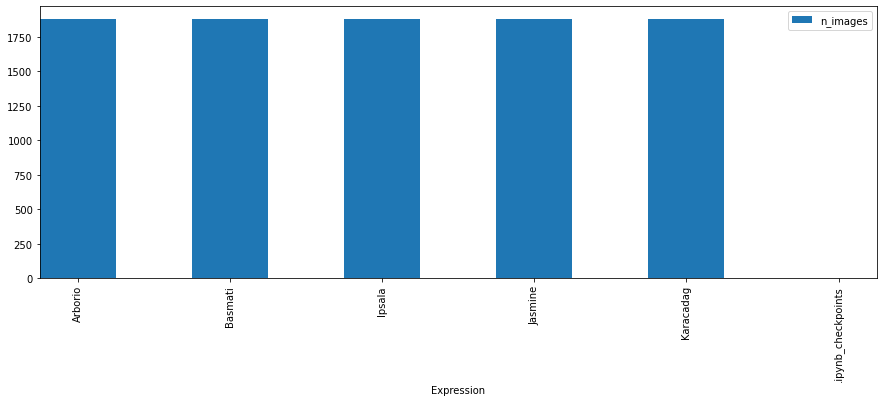
Arborio Basmati Ipsala Jasmine Karacadag

* 15K images for each type of rice grain
* Task: Rice grain classification

# Sampling

Target classes are equidistributed. 75K images → 15K images for

each type of rice

Sampling to train NN from scratch and pre-trained NN with transfer learning

5K images → 1K images for each type of rice

Sampling to train best NN from scratch and best pre-trained NN

with transfer learning

* Step to define a NN from scratch
* Pre-processing + Splitting data into training and validation: 80% of the data is contained in training set, 20% of the data is contained in the validation set

Pre-processing for training set: standardization + data augmentation

Pre-processing for validation set: standardization

10

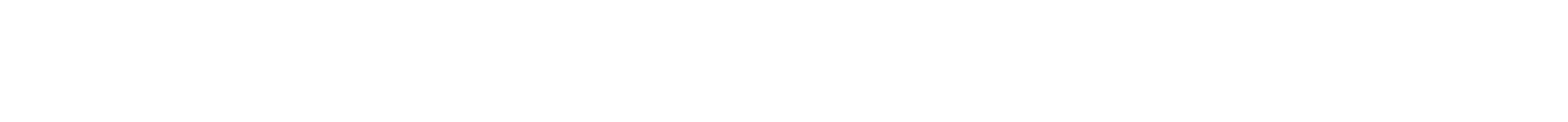
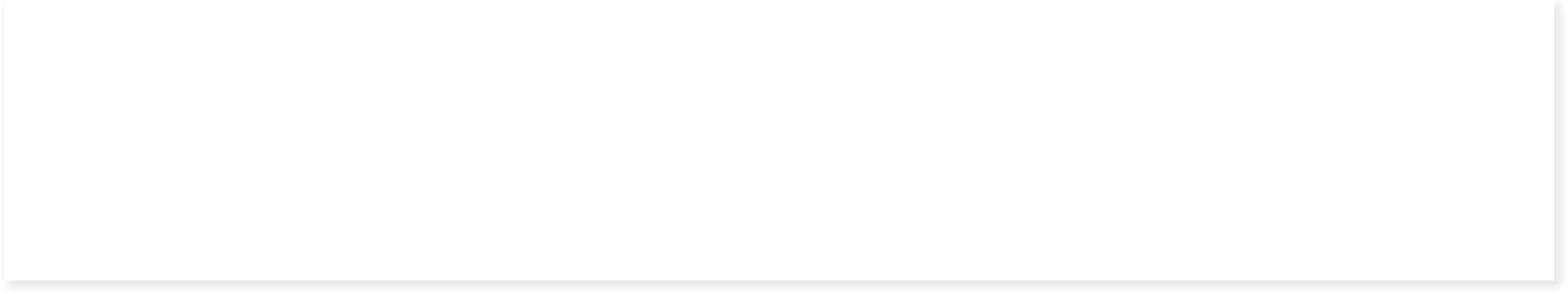
K images

→

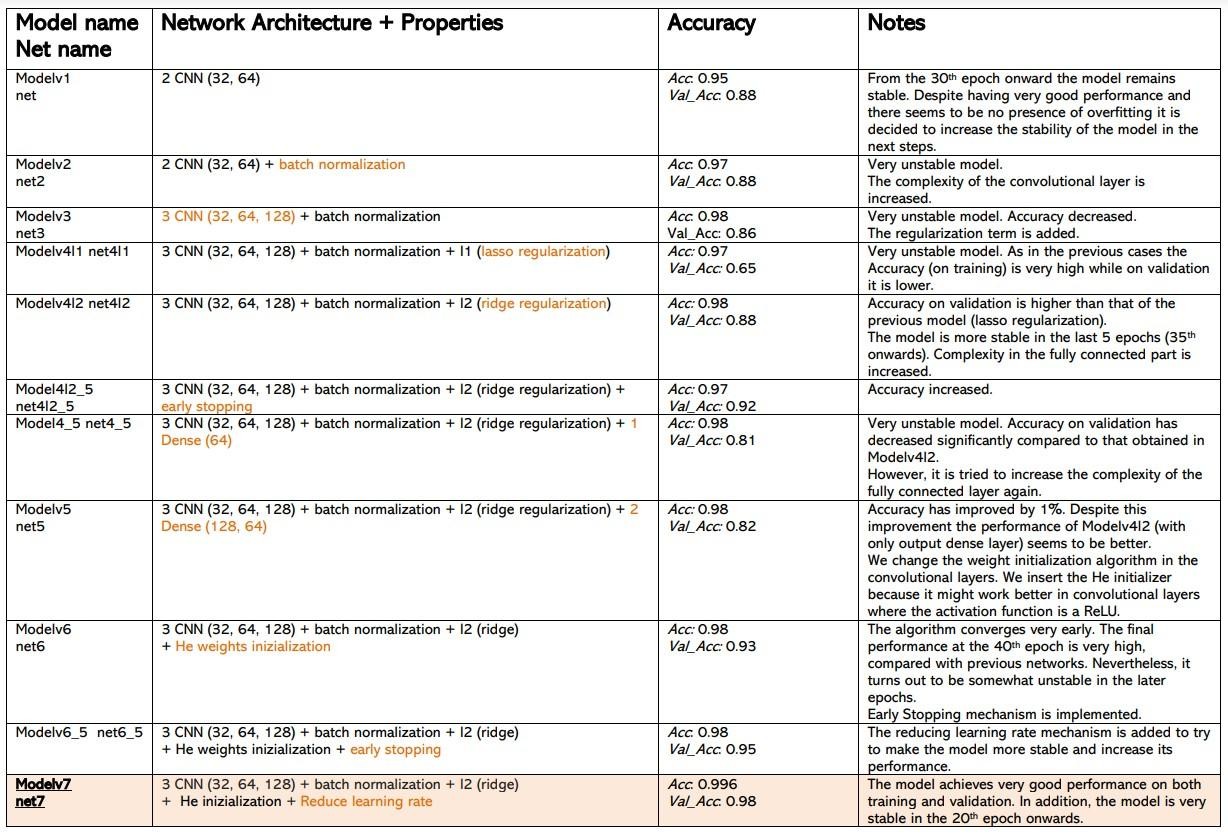
K images for

2

each type of rice



* NN Architecture definition
* NN Compilation: definition of network macro-elements (optimizer and learning rate, loss function, additional metrics, parameters, etc.)
* NN Training
* Summary and Results visualization

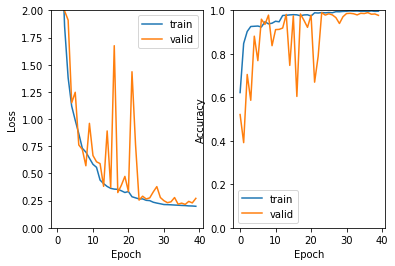


Best Model

## Architecture

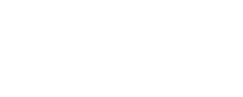
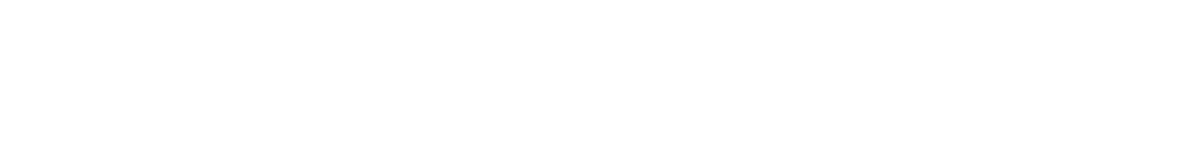
* Input Layer
* 2D convolutional layer: 3x3 filter, 32 neurons, Ridge

regularizer, He uniform inizializer Training and Training and

* ReLU activation function Validation Loss Validation Accuracy
* Batch Normalization layer
* Max Pooling layer (3x3)
* 2D convolutional layer: 3x3 filter, 64 neurons, Ridge regularizer, He uniform inizializer
* ReLU activation function
* Batch Normalization layer
* Max Pooling layer (3x3)
* 2D convolutional layer: 3x3 filter, 128 neurons, Ridge regularizer, He uniform inizializer
* ReLU activation
* Batch Normalization layer ● Global Max Pooling layer
* Output layer: Softmax activation function, Ridge regularizer

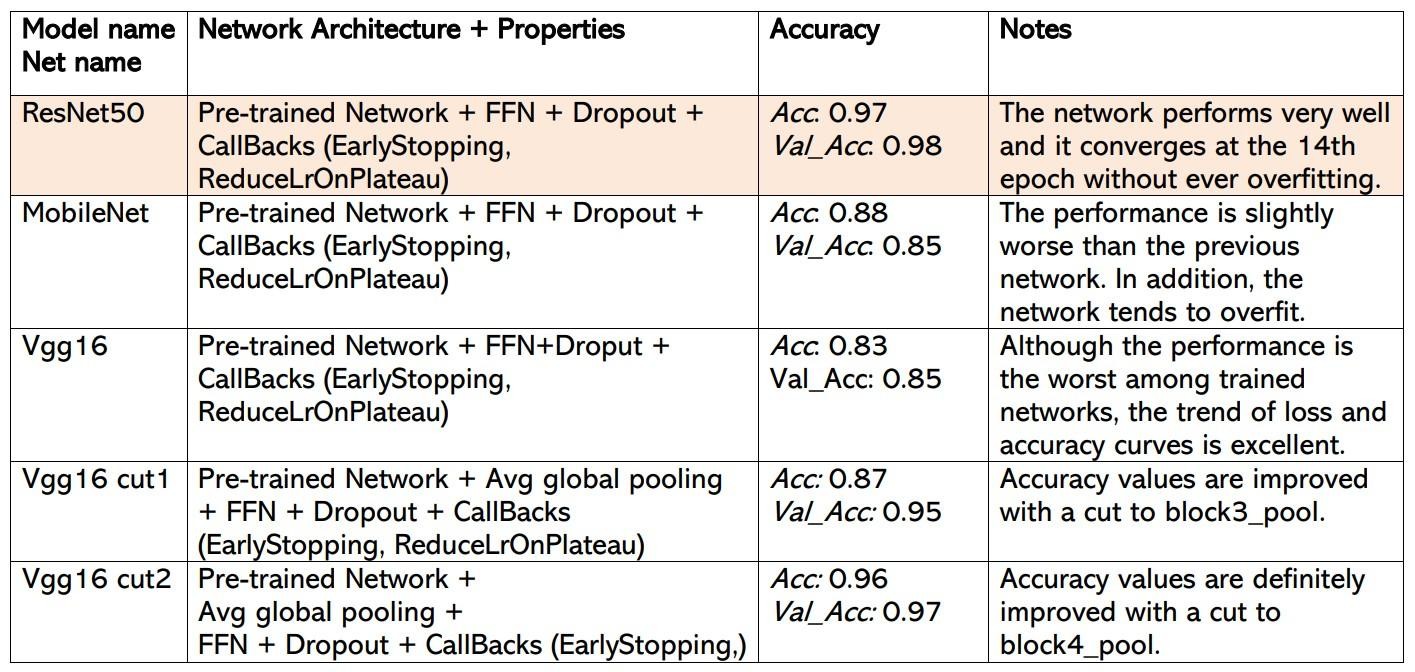
Callback: Reduce Learning Rate

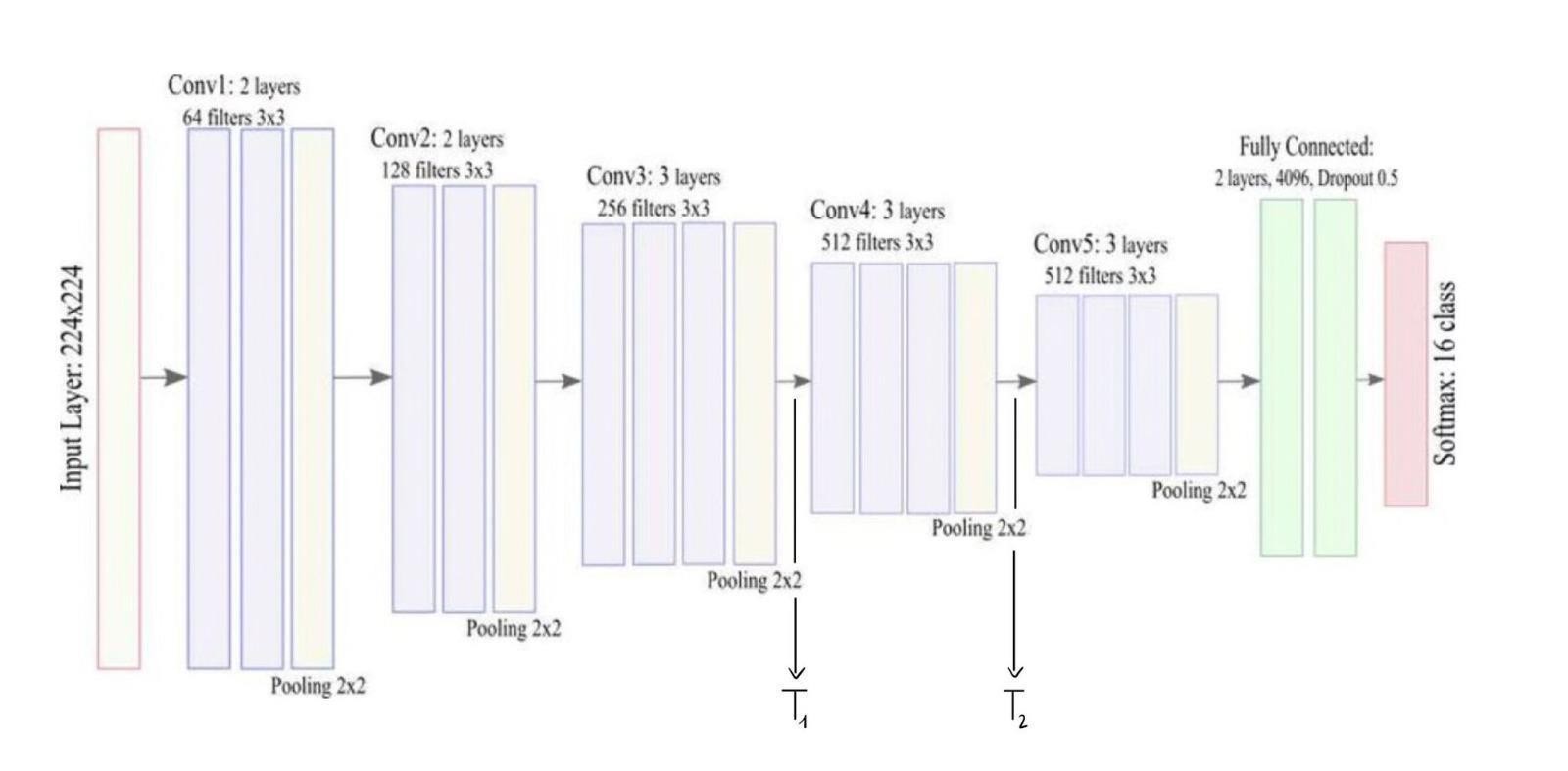
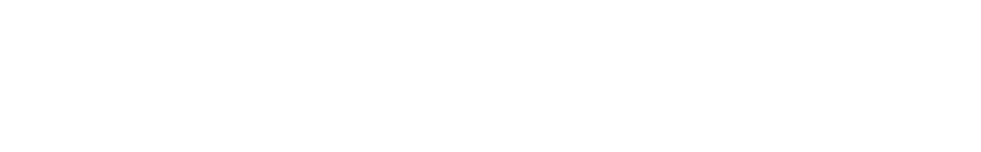
* + Preprocess Data: The images are converted from RGB to BGR, then each color channel is zero-centered with respect to the ImageNet dataset, without scaling.

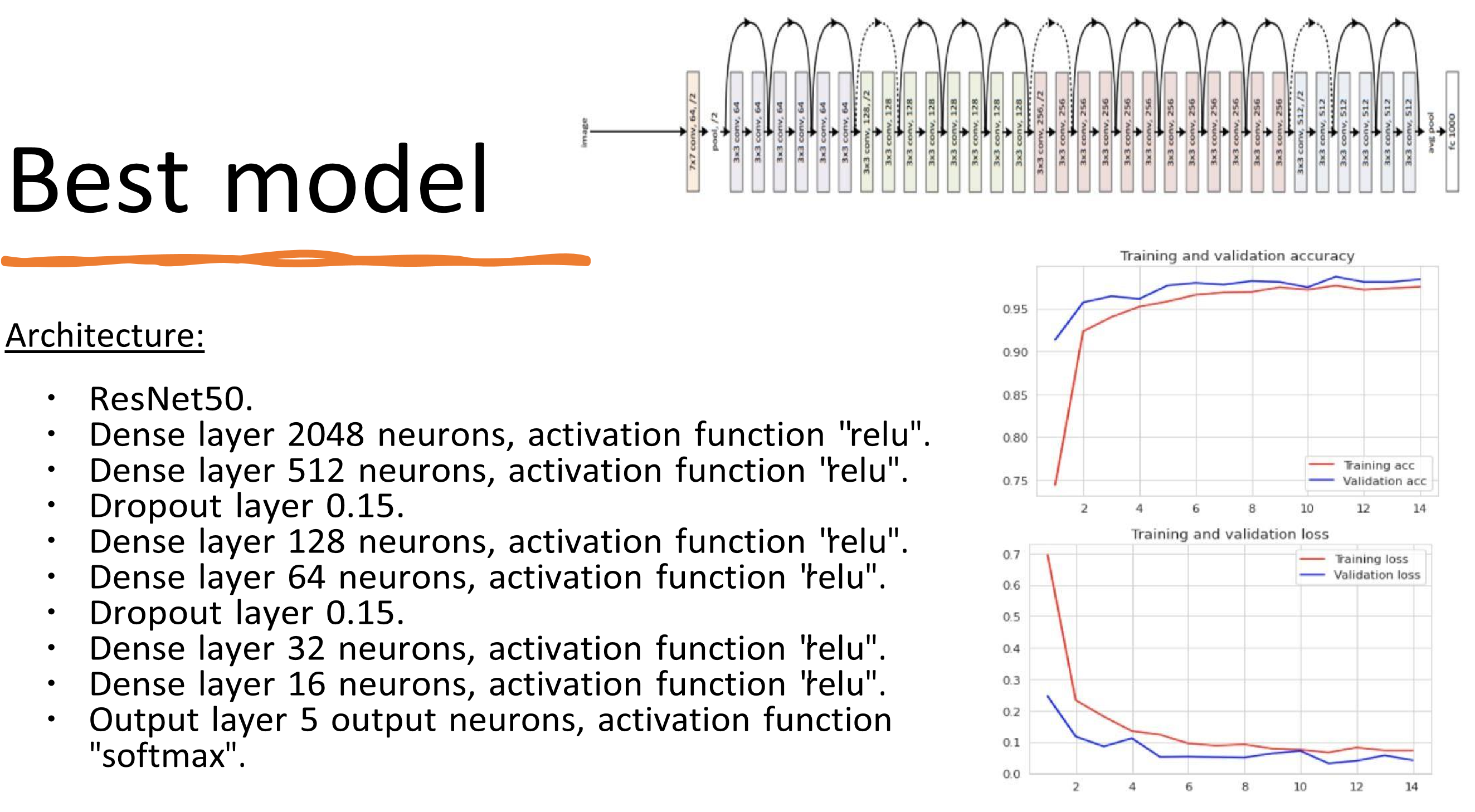


* + Loading of the pre-trained network.
  + Freeze the convolutional base before compiling and train the model.
  + Define a FNN architecture.
  + Callbacks.
  + Compile and train.

Applications Results

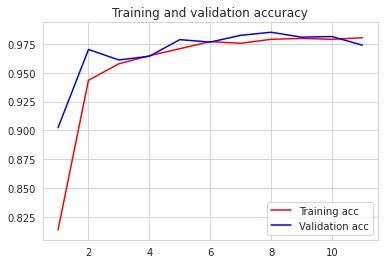
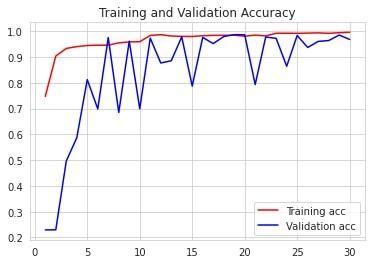
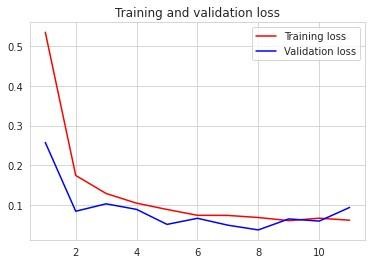
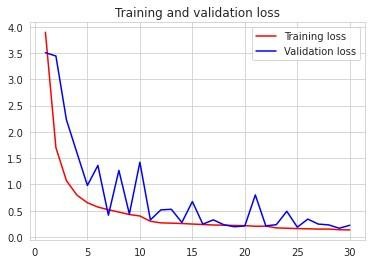






Best Models Evaluation on 10k dataset

Best Model from Scratch



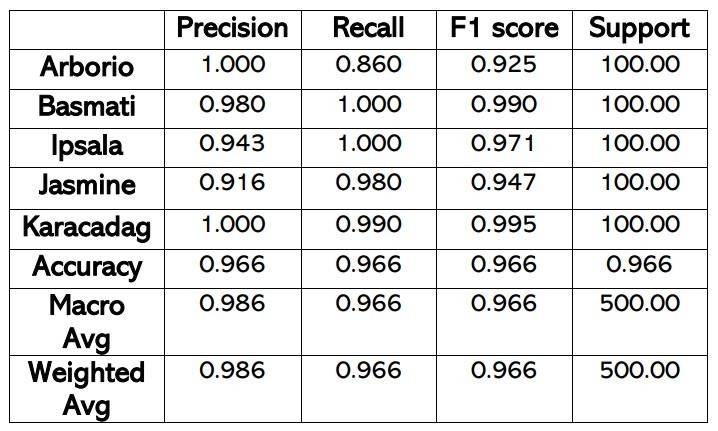
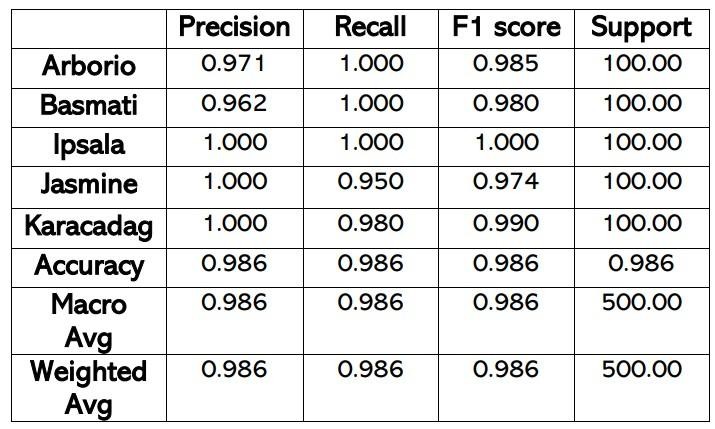
Best Model with

Transfer Learning

# Classification Report on Test set

Best Model from scratch

Best Model with Transfer Learning



Thanks for your attention