UNIVERSITÀ DEGLI STUDI DI MILANO BICOCCA

FLOWER RECOGNIZER FOR MOBILE APPLICATION

LORENZO PIROLA - MAT. 816418
MATTEO ROMANATO - MAT. 816852
YOUSSEF KARRATI - MAT. 817435

Outline

1. Introduction
2. Dataset
3. Models
4. Models compression
5. Conclusions

Outline

1. **Introduction**
2. Dataset
3. Models
4. Models compression
5. Conclusions

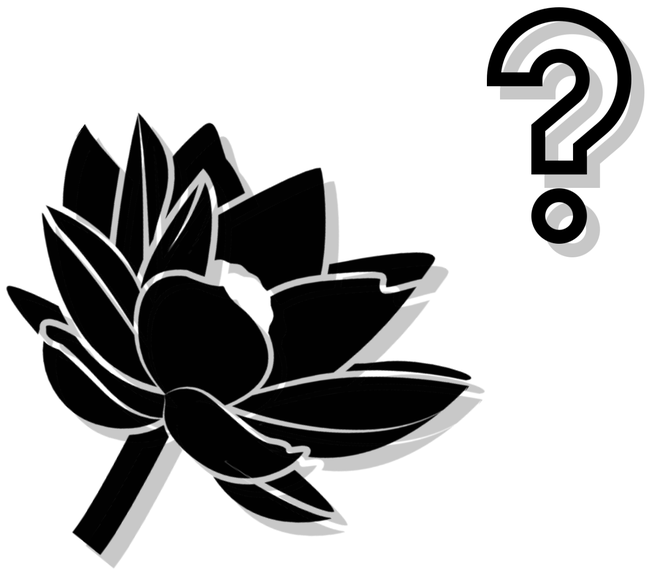
Advanced Machine Learning

Introduction

Objective

The aim of the **project** was to create a model suitable to a **mobile context** that can recognize **flowers** from images.

For the creation of the model three different architectures were chosen for comparison as starting point: MobileNetV2, DenseNet-121 and EfficientNet-B0.

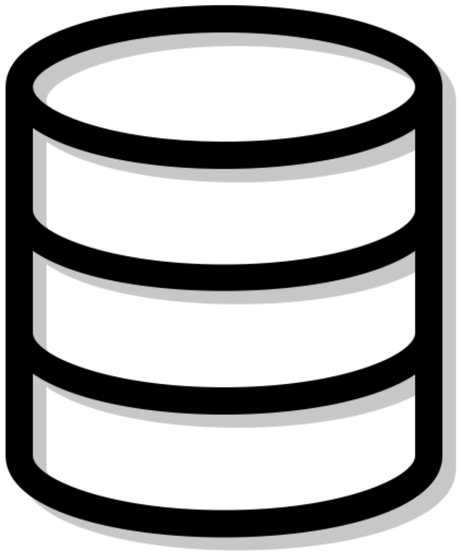


Outline

1. Introduction
2. **Dataset**
3. Models
4. Models compression
5. Conclusions

Advanced Machine Learning

Dataset



The **dataset** used contains 8189 images of flowers commonly found in the UK, belonging to 102 different classes. It was divided into:

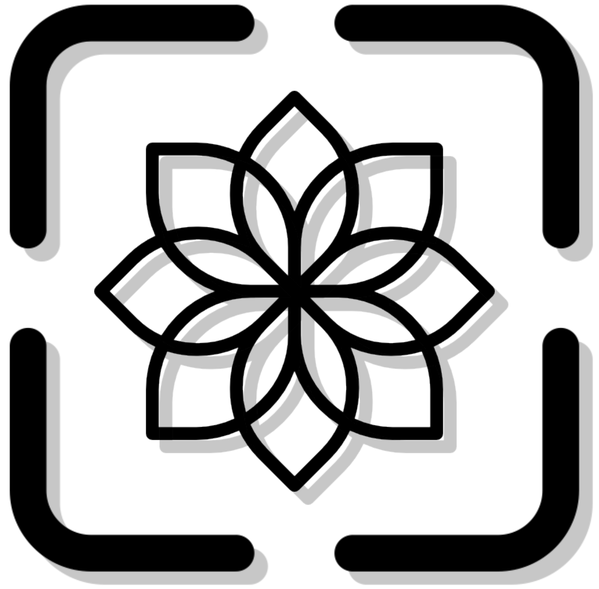
- **Train set** using 80% of the available images (5284)
- **Test set** using the remaining 20% of the images (1638)
- **Validation test** was created using 20% of train set images (1267)

Advanced Machine Learning

Preprocessing

Normalization

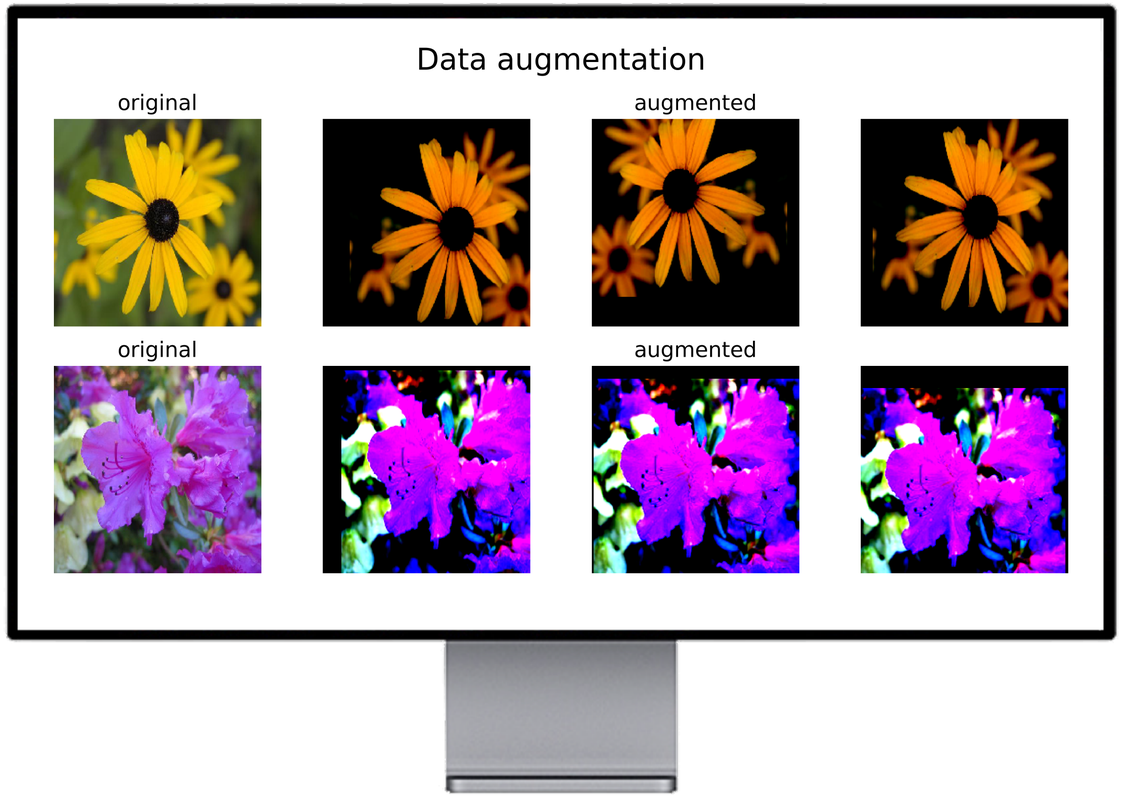
- **MobileNetV2:** rescales pixels in a range between -1 and 1
- **DenseNet-121**: rescales pixels in a range between 0 and 1 and normalizes channels with respect to the ImageNet dataset
- **EfficientNet-B0**: converts from RGB to BGR and normalizes channels with respect to the ImageNet dataset



Preprocessing

Data augmentation

Width shifting and **height shifting**



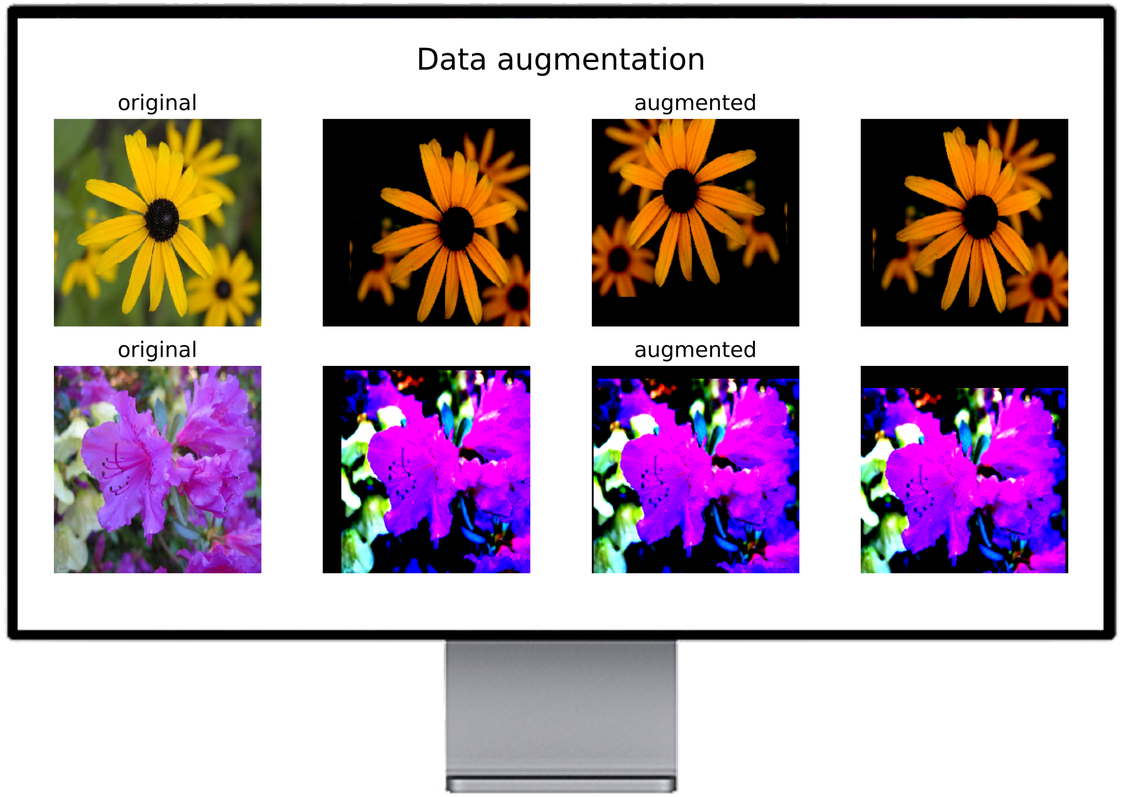
Advanced Machine Learning

Preprocessing

Data augmentation

Width shifting and **height shifting**

Horizontal flipping



Advanced Machine Learning

Preprocessing

Data augmentation

Width shifting and **height shifting**

Horizontal flipping

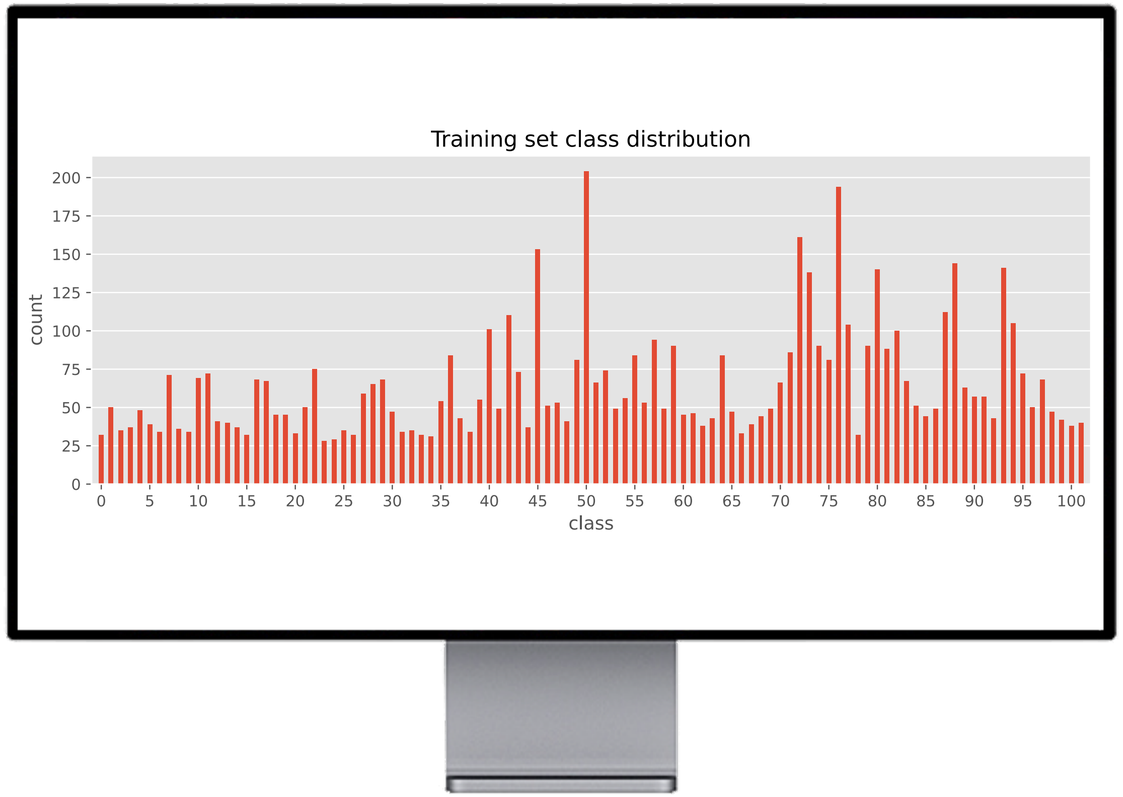
Brightness

Advanced Machine Learning

Preprocessing

Class weight balancing

Weight-balancing approach in the loss function to overcome the problem of **unbalanced** dataset.



Outline

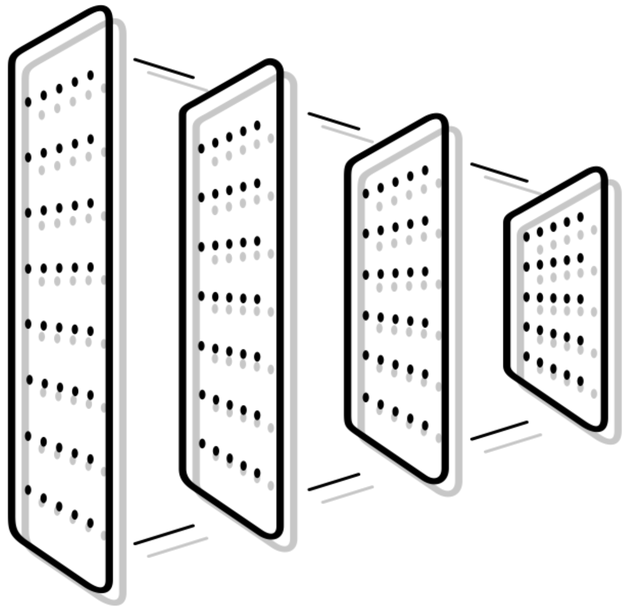
1. Introduction
2. Dataset
3. **Models**
4. Models compression
5. Conclusions

Advanced Machine Learning

Models

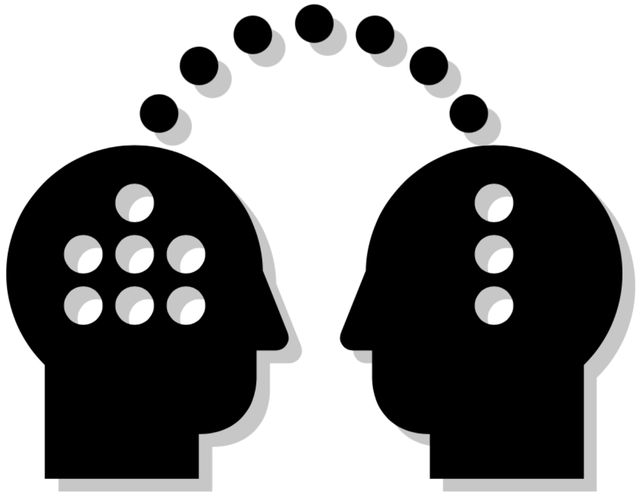
Architectures

- MobileNetV2
- DenseNet-121
- EfficientNet-B0



Advanced Machine Learning

Models



Transfer learning

It was used a **fine-tuning** techniques due to the features of the dataset.

Cut on the **last layer** provided the best performance.

Advanced Machine Learning

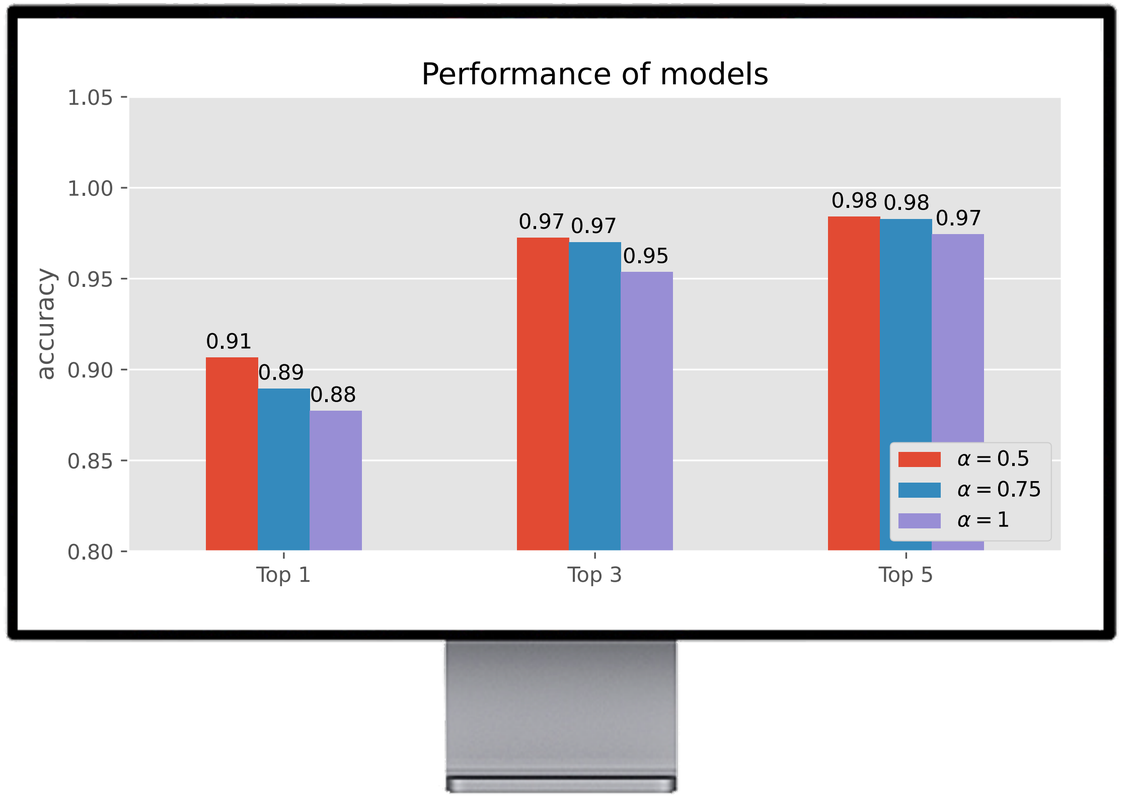
Models

MobileNetV2 comparison

MobileNet allowed to handle network width through a **parameter** α.

Parameter values tested:

- α = 0.5
- α = 0.75
- α = 1

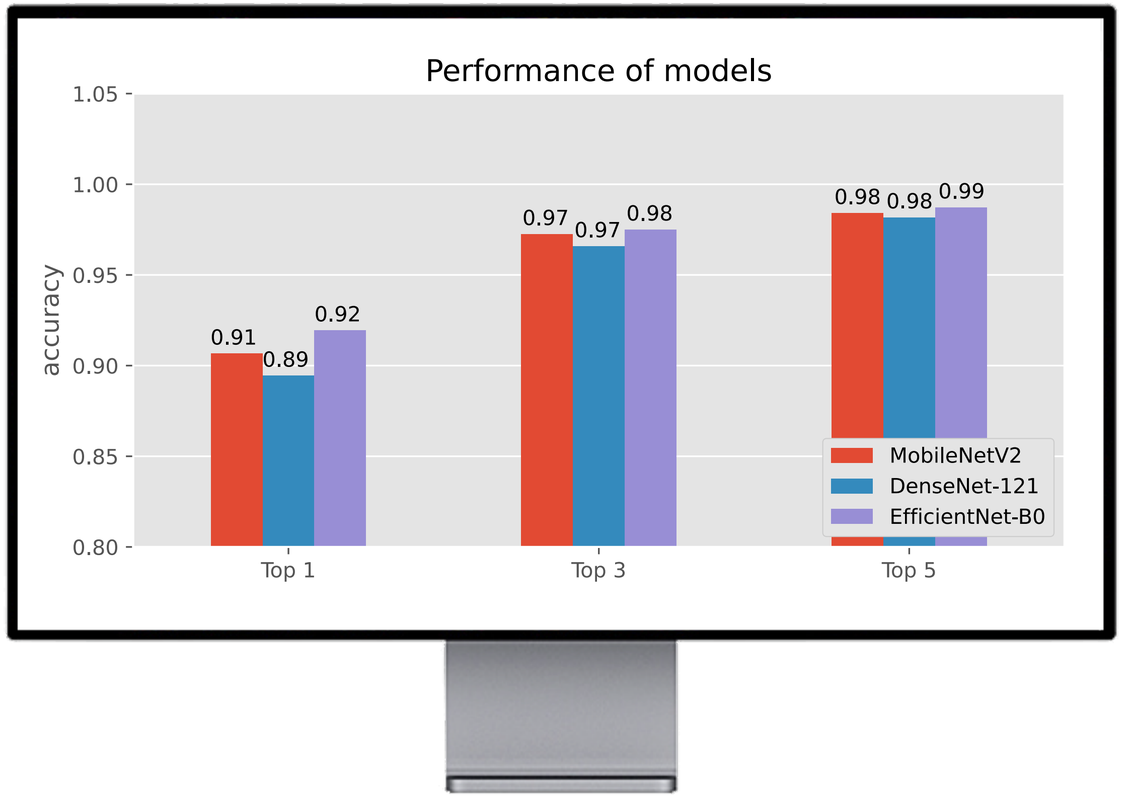


Advanced Machine Learning

Models

Accuracy comparisons

The **final comparison** brought good performances of the three model but in terms of accuracy EfficientNet was the best.



Outline

1. Introduction
2. Dataset
3. Models
4. **Models compression**
5. Conclusions

Advanced Machine Learning

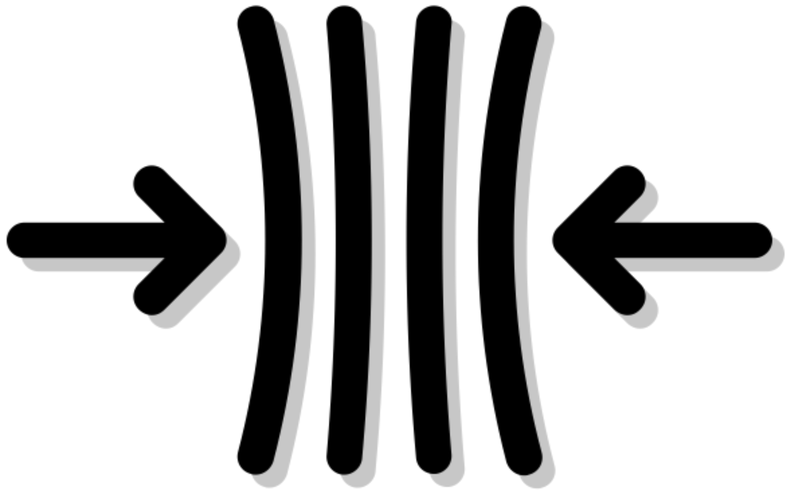
Models compression

Iterative pruning

- Increasing sparsity through iterations
- Using **Keras magnitude-based pruning**
- On **convolutional** (normal and pointwise) and **fully connected** layers

Advanced Machine Learning

Models compression



Quantization

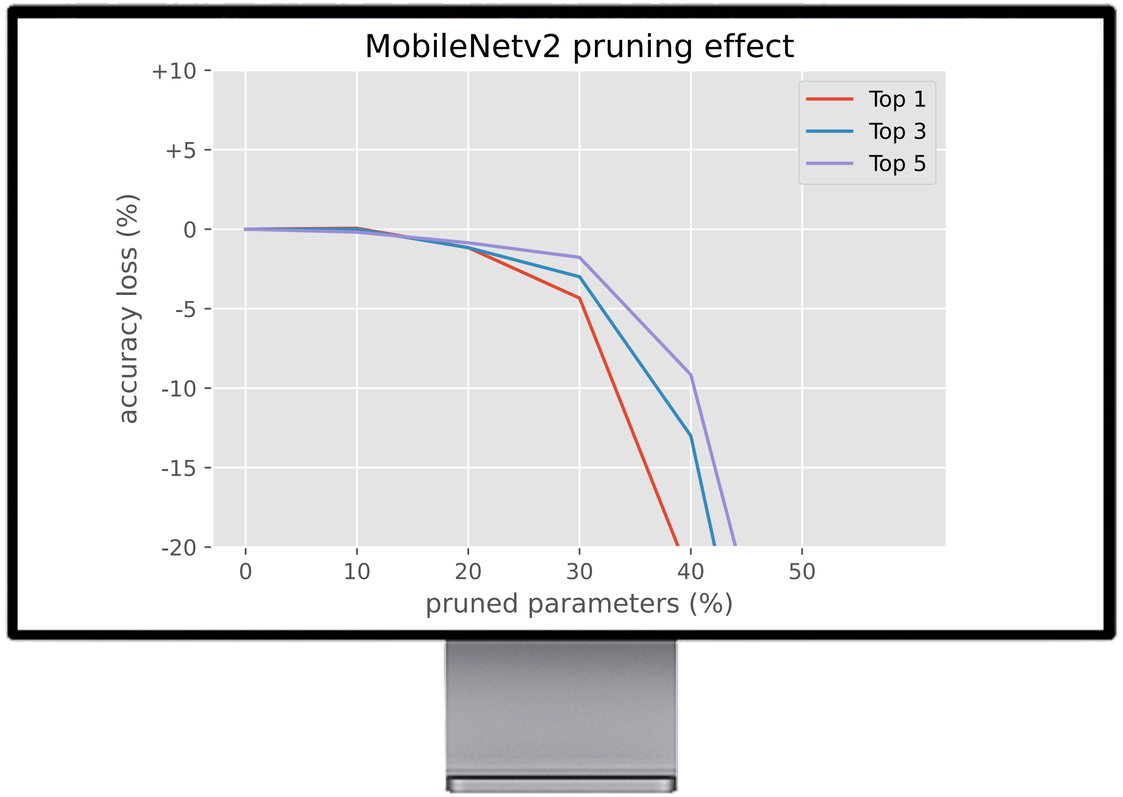
- Post training quantization using **TensorFlow Lite**
- **Dynamic range quantization** for mobile
- **Float16 quantization** for GPU

Advanced Machine Learning

Models compression

MobileNetV2

- **Sparsity**: 20%
- **Accuracy**: from 90.66% to 89.82%

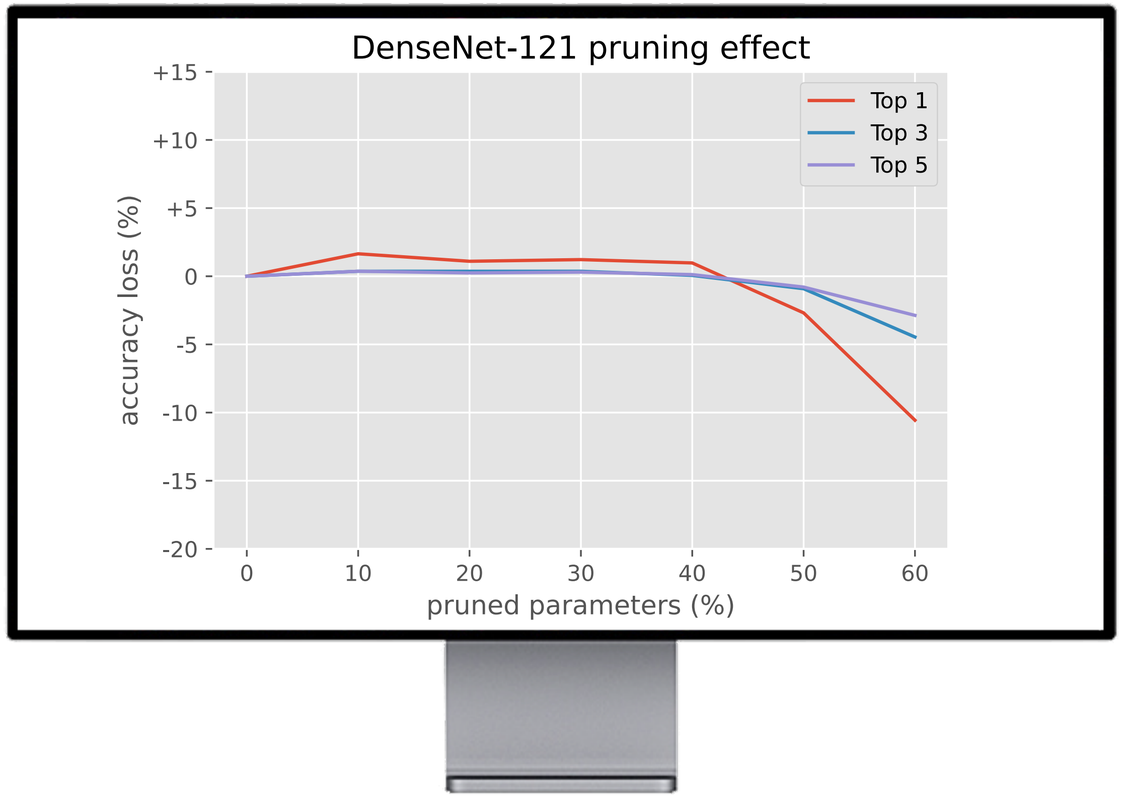


Advanced Machine Learning

Models compression

DenseNet

- **Sparsity**: 40%
- **Accuracy**: from 89.44% to 90.42%

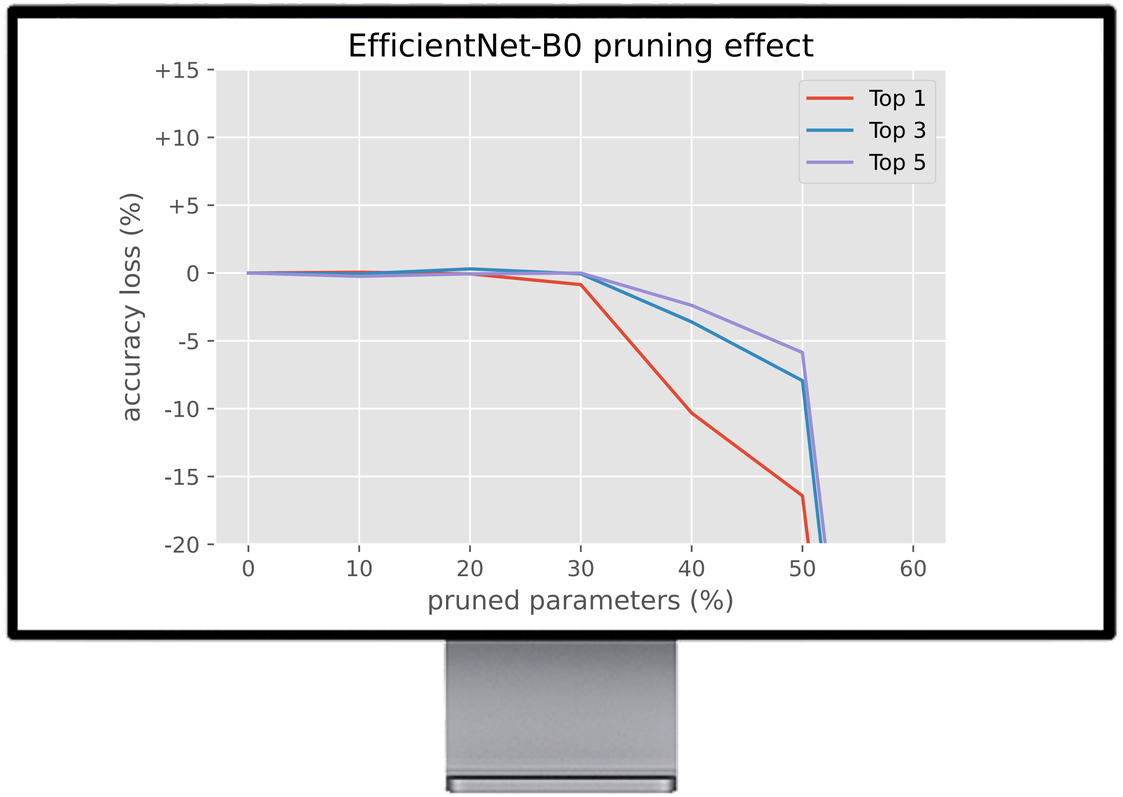


Advanced Machine Learning

Models compression

EfficientNet

- **Sparsity**: 30%
- **Accuracy**: from 91.94% to 91.09%

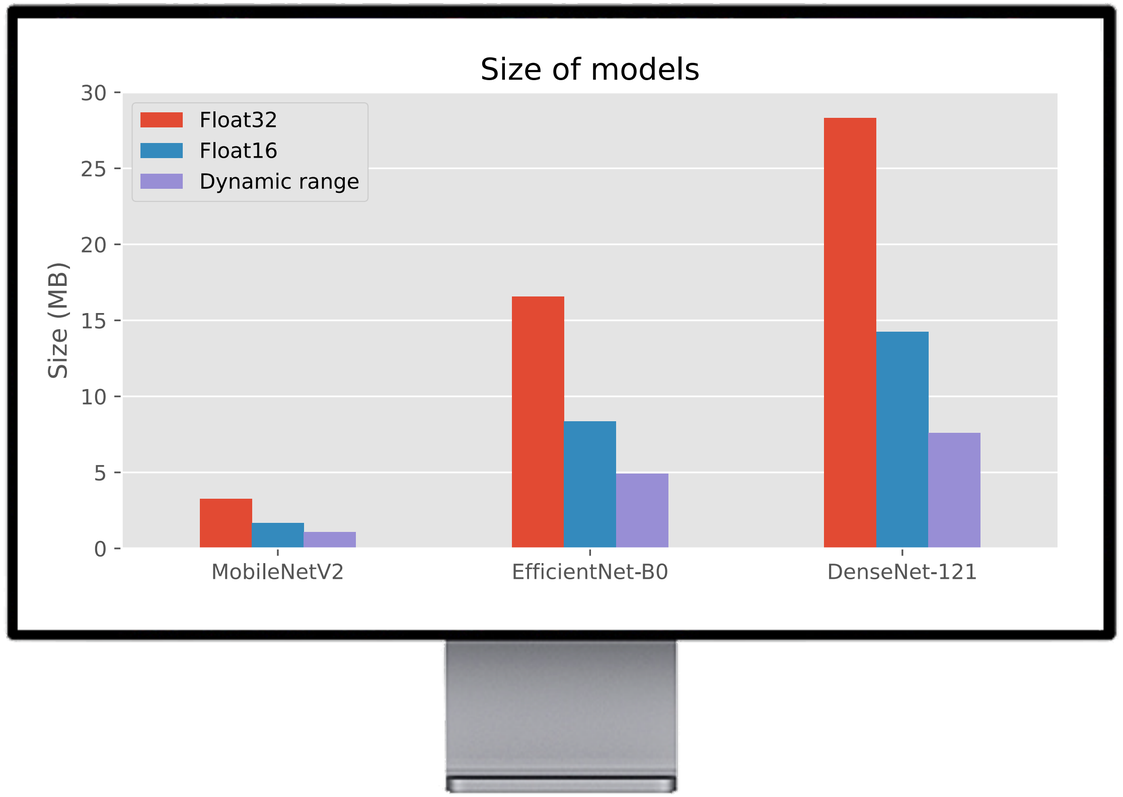


Advanced Machine Learning

Models compression

Quantization

- **MobileNetV2**
  - **Size**: from 3.27 MB to 1.07 MB (x3.06)
  - **Accuracy**: from 89.50% to 89.26%
- **EfficientNet**
  - **Size**: from 16.56 MB to 4.92 MB (x3.37)
  - **Accuracy**: from 91.09% to 91.33%
- **DenseNet**
  - **Size**: from 28.31 MB to 7.60 MB (x3.72)
  - **Accuracy**: from 90.42% to —



Outline

1. Introduction
2. Dataset
3. Models
4. Models compression
5. **Conclusions**

Advanced Machine Learning

Conclusions

Results and future studies

- Model based on EfficientNet with 91.33% accuracy and size 4.92 MB

- Investigating impact of data augmentation on performance

References

- M.-E. Nilsback, “An automatic visual Flora – segmentation and classification of flowers images,” Ph.D. dissertation, University of Oxford, 2009.
- M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, “Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation,” CoRR, vol. abs/1801.04381, 2018.
- A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” CoRR, vol. abs/1704.04861, 2017.
- G. Huang, Z. Liu, and K. Q. Weinberger, “Densely connected convolutional networks,” CoRR, vol. abs/1608.06993, 2016.
- M. Tan and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” CoRR, vol. abs/1905.11946, 2019.
- I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, Deep learning. MIT press Cambridge, 2016, vol. 1, no. 2, ch. 7.12, p. 253.

THANK YOU