ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

BUSINESS ANALYTICS Master of Science

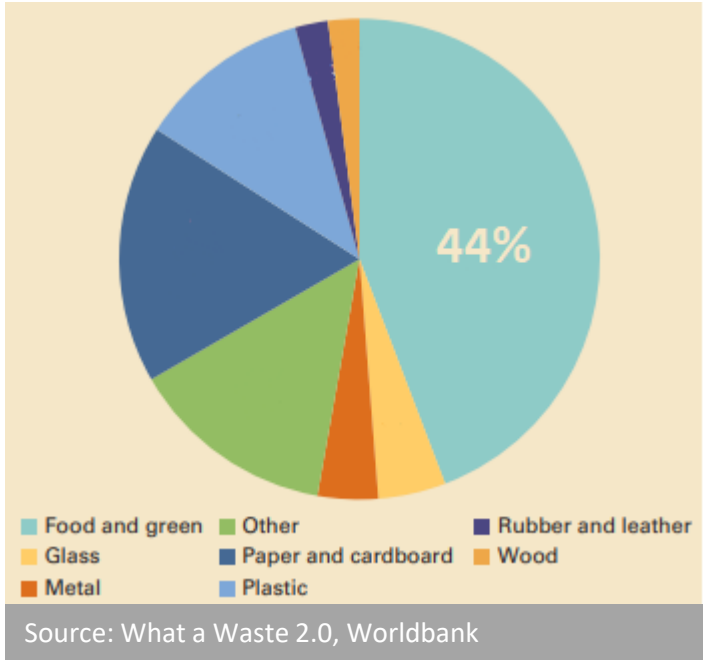
# MLCA Mini-Project: Identifying recyclable/organic waste

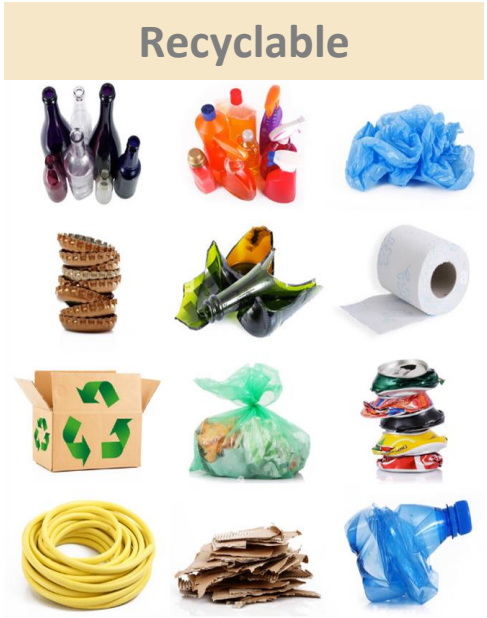
Professor: Haris Papageorgiou
Students: Michail Kalligas, F2822103
Panagiotis Iolos, F2822105

Athens, 2022

# The Business Problem

- Recycling has become ever more important
- 5% projected growth on a yearly basis until 2028
- Technology has made recycling easier in various ways
- Separation of organic from recyclable waste is done manually




# Proposed Solution

**1) Identify organic waste during the filtering process of recyclable waste**

2) Categorize organic waste instead of removing it

3) Deliver each type of organic waste to an appropriate plant

The purpose of this project is to automate step 1 of the above by:

- Utilizing TensorFlow's MobileNetV2 image recognition model
- Using transfer learning to fine-tune the model for the classification of organic and recyclable waste

# The Dataset

- Waste Classification Data by Sashaank Sekar
- Obtained from Kaggle
- 25077 images
- 56% organic, 44% recyclable
- 85% training, 15% testing

# MobileNetV2

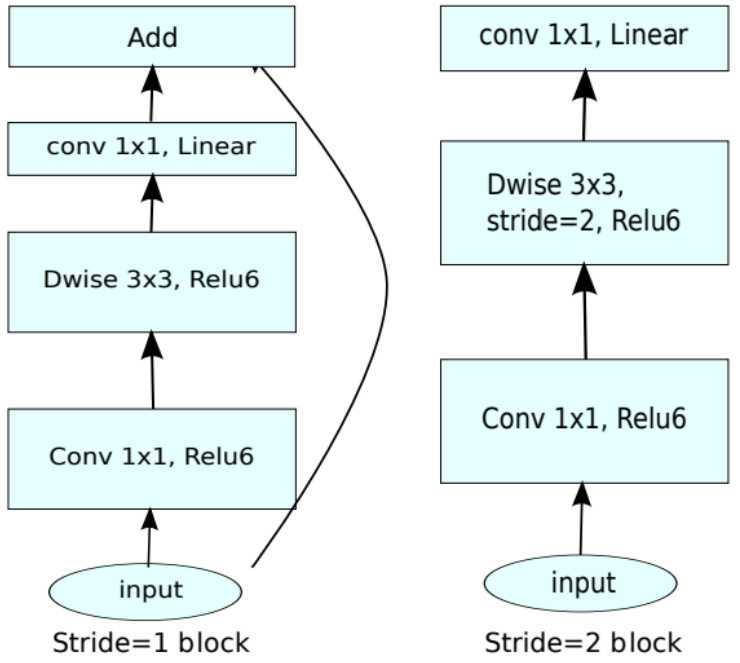
A convolutional neural network pretrained on the ImageNet Dataset, a dataset containing over 1.4 million images spread among 1000 classes.

Implements:

- Depthwise separable convolutions
- Linear bottlenecks
- Inverted residual blocks

Pretrained for:

- Image classification
- Object detection
- Semantic segmentation



# Feature extraction

Pipeline:

- Reshaping inputs
- Data augmentation layer
- Preprocessing layer
- MobileNetV2 base pretrained model (frozen weights)
- Global average pooling layer
- Dropout layer
- Dense prediction layer

```
model.summary()
```

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 160, 160, 3)]     0

 sequential (Sequential)     (None, 160, 160, 3)       0

 tf.math.truediv (TFOpLambda  (None, 160, 160, 3)      0
 )

 tf.math.subtract (TFOpLambd  (None, 160, 160, 3)      0
 a)

 mobilenetv2_1.00_160 (Funct  (None, 5, 5, 1280)       2257984
 ional)

 global_average_pooling2d (G  (None, 1280)             0
 lobalAveragePooling2D)

 dropout (Dropout)           (None, 1280)              0

 dense (Dense)               (None, 1)                 1281

=================================================================
Total params: 2,259,265
Trainable params: 1,281
Non-trainable params: 2,257,984
_________________________________________________________________
```

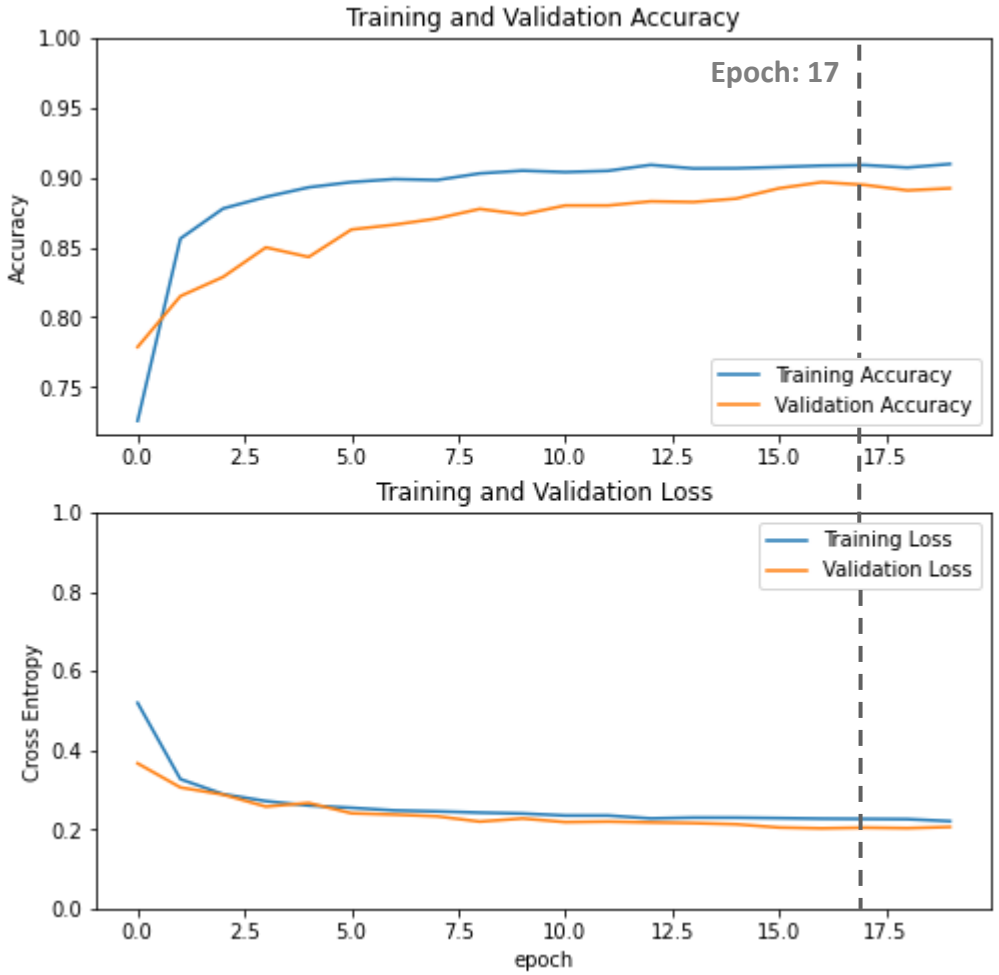
# Training and results

Parameters

- Loss function:Binary Cross-Entropy
- Learning rate: 0.0001
- 20 Epochs

Results

- Optimal number of epochs: 17
- Accuracy: ~90%
- Loss: 0.2



# Fine-tuning

- Unfroze 56 layers of MobileNetV2
- Trainable Parameters: 1,862,721
- Lowered learning rate

```
model.summary()
```

```
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_2 (InputLayer)        [(None, 160, 160, 3)]     0

 sequential (Sequential)     (None, 160, 160, 3)       0

 tf.math.truediv (TFOpLambda  (None, 160, 160, 3)      0
 )

 tf.math.subtract (TFOpLambd  (None, 160, 160, 3)      0
 a)

 mobilenetv2_1.00_160 (Funct  (None, 5, 5, 1280)       2257984
 ional)

 global_average_pooling2d (G  (None, 1280)             0
 lobalAveragePooling2D)

 dropout (Dropout)           (None, 1280)              0

 dense (Dense)               (None, 1)                 1281

=================================================================
Total params: 2,259,265
Trainable params: 1,862,721
Non-trainable params: 396,544
_________________________________________________________________
```

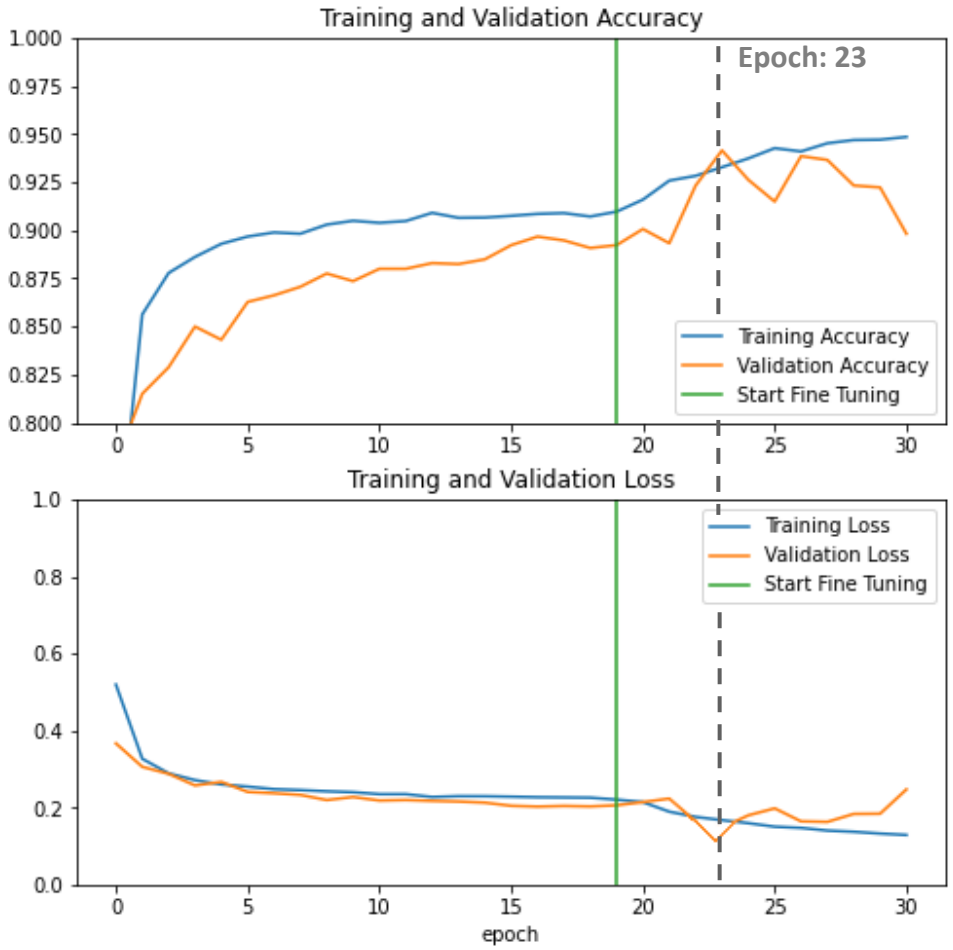
# Training and results

Parameters

- Loss function: Binary Cross-Entropy
- Learning rate: 0.00001
- 10 Epochs

Results

- Optimal number of epochs: 23
- Validation accuracy: ~95%
- Validation loss: ~0.14



# Predictions

O

R

R

O

O


R

R

O

O

R


O

R

R

O

O

O

R

O

# Qualitative & Error Analysis

Recyclable

Organic

Recyclable

Recyclable

Recyclable

Recyclable

Recyclable

Recyclable

Recyclable

# Method shortcomings & Future work

Limitations:

- Limited knowledge in sciences related to the recycling industry
- Business need for a lightweight, easily mountable system
- Limitations concerning the variety and number of items in the dataset

Up next:

- Binary Classifier → Multiclass Classifier
- Model deployment in a real sorting line of a recycling plant
- Try more complex architectures (Microsoft’s Swin, Google’s ViT)

# “Waste isn’t waste until we waste it”