Student ID: 104359382

Name: Hong Anh Nguyen

## Question 1: Explain the concept of transfer learning. How does it differ from training a model from scratch

* Transfer learning introduces ideas using a model which trained on a **large** corpus of data with general and wide range of types, more important, they were trained with a proper ***receipt*** *of hyper-parameter* making these model have a high performance then these models are used for a more **downstream task** specifying for user need.
* The key point here is that we take advantage of **general knowledge** and the high performance of these **pre-trained** models, using it for our specific task instead of training a whole new model from scratch with our dataset which requires us to hyper-parameter tuning and train on a large enough epoch.

## Question 2: What is fine-tuning in the context of transfer learning, and why is it useful ?

* Fine-tuning refers to a technique which only trains a part of the pre-trained weights, making it adapt to new tasks with reasonable effort. Fine-tuning reduces the computational cost since it only trains a small number of pre-trained weights instead of training the whole weights which is very useful for the real-world application (many big techs are already provided with lots of large pre-trained models).
* The are several methods to apply fine-tuning, the most straightforward is freezing most of pre-trained weight and only leaving some weight at the top (the very last layer) to train. Other proficient techniques people use are LoRA (LowRank Adaptation), adapter…

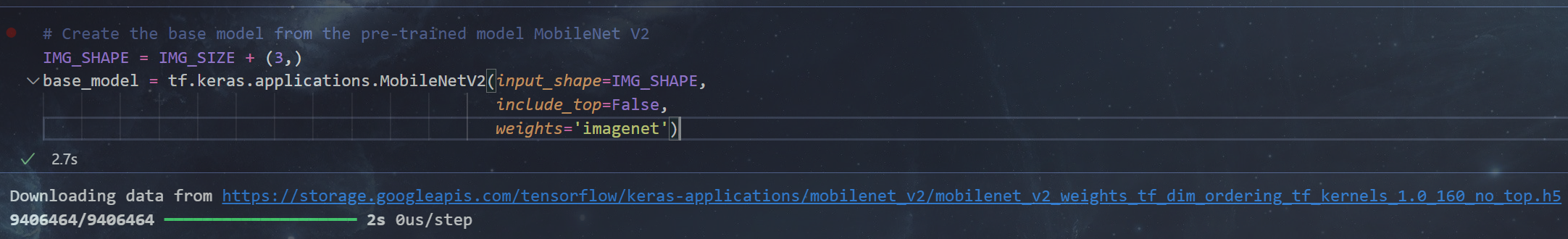
## Question 3: Why is it important to freeze the convolutional base during feature extraction?

* Since you want to take advantage of the high-quality **knowledge** of the pre-trained model which stored in their weight, it is important to freeze that knowledge during the feature extraction to make sure these weights stay unchanged and not being messed up which normally leads to reduce accuracy.

## Question 4: Why data augmentation?

* This is extremely important in case the dataset is small which potentially causes **overfitting.** This is likely caused by small dataset might only contain several poses, position of view, color, brightness – image attributes of object leading to the model can only predict correctly if the new sample have some attributes same with these data. Data augmentation dealing with this by changing these attributes randomly from the original data making it more general, represent better the real-world data and reduce overfitting.

## Screen shot of the code snippet where the pre-trained MobileNetV2 model is loaded without the top classification layers.



## Screen shot of the portion of code where the pre-trained model is set to be non-trainable for feature extraction purposes.

A close up of a computer screen

AI-generated content may be incorrect.

## Screen shot of the data augmentation layers defined in the model.

A screenshot of a computer program

AI-generated content may be incorrect.

## Screen shot of the code that shows the addition of the new classifier layers on top of the base model.

