GOAL: Recognise and separate 5 classes items in images

In cases, when needed to classify images into couple classes and having not a lot of data, a very often used strategy is to use an algorithm pretrained on a very big image set and retrain only couple layers. Because of the lack of computer power, this model should have the highest possible accuracy with lowest size, number of parameters and depth. Using a table of some models available in Keras library, <https://keras.io/applications/>, Xception model looked to fit chosen criteria the most. ‘Imagenet’ was used to pretrain Xception model weights, it is one of the biggest and definitely the most popular image library, containing over 10 million of classified images.

System runs by starting ‘Xception2.py’ file, which downloads training images from the list saved in the file ‘train\_products.csv’ (some saved links were failing), resizes them into the same size of 256\*256, divides into validation and training sets, saves them into specific category folders and fine tunes the model’s weights called ‘top\_model\_weights.h5’. Then, ‘prediction.py’ file is run to download test images from the list saved in the file ‘test\_products.csv’, predicting their content’s category and saving results into a ‘test\_products\_predictions.csv’ file.

Sadly the lack of computational power and limited time to create the model did not allow achieving the true potential accuracy of the model. A very limited number of training iterations was possible, which did not allow finding best hyperparameters. For example, it took around 15 hours of training the model to achieve around 65% accuracy on validation data set using only one fifth of the whole available training data after around 15 epochs. There is a big potential to achieve, for example, by choosing different learning rates or using the whole data set.

FUTURE: Next step is to run the model on Google Cloud platform, which provides relatively cheap, but very powerful computational power ‘for rent’, and continue looking for better hyperparameters to achieve higher accuracy.