Edit Share

Places365 classification

Project number 1

Kacper Puchalski, Jacek Kowalski, Karol Guzikowski

Introduction

As specified earlier in the semester, we are tackling a classification task. We were supposed to prepare a fully deveolped project using Kedro framework.

Data

We selected Places365 as our dataset. It consists of photos presenting a specific scenario, like beach, airport, indoor bar, etc. roughly 365 different classes. There are 18 milion train images all together, 50 images per class in validation set and further 900 pictures per scene for testing. Major data is given as images in high resolution. However, we do not have the need for such detailed representation, so we made use of available version of dataset with 256x256 pictures.

We prunned the data to fit our needs, leaving 1500-500-500 images respectively for train, validation and test sets. It let us fit the data into 20GB kaggle memory limit for training purposes.

Model

After some thought on model architecture, we decided to go with one already pretrained on ImagNet. MobileNetV3 Small was our choice, as it is complex enough to solve problem of cassifying different scenarios whilst simultaneously being quite small to fit our computational capabilities.

In order to apply MobileNet on our dataset we will modify and retrain the classifier layers.

Training

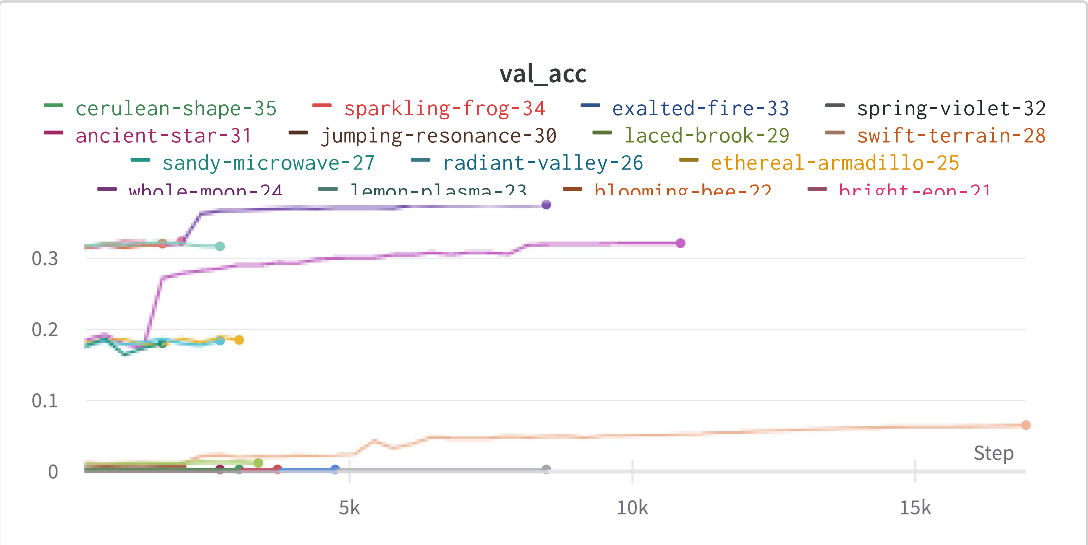
Using Pytorch Lightning alongside Kedro we created a training pipeline. Training is perfromed using Adam optimizer with a learning rate scheduler, which reduces lr when cross entropy loss on validation set stops improving. Furthermore, we applied early stopping, so training process would not last forever nor allow overfitting.

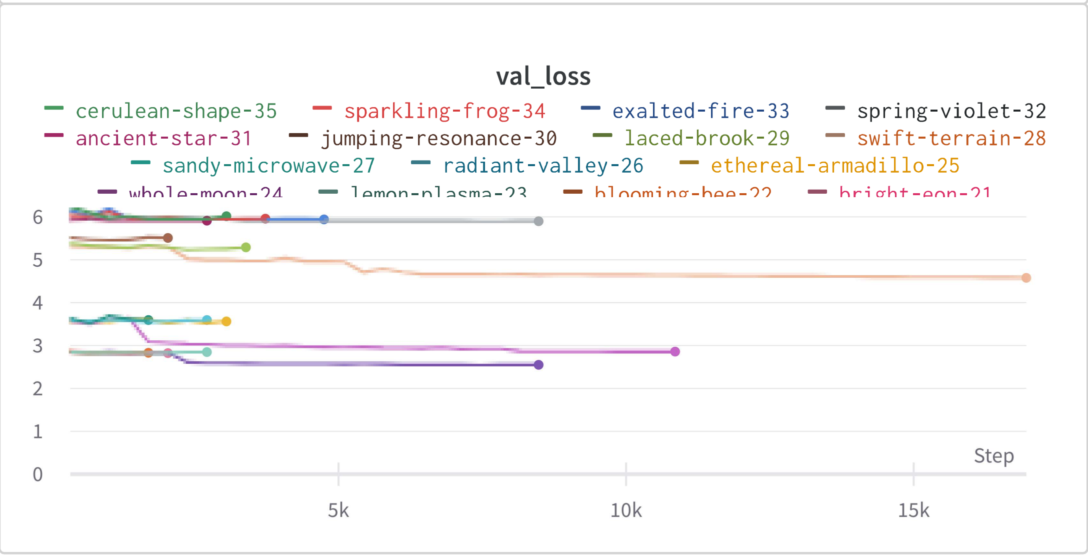
Training was performed on a single GPU.

Grid Search

We came to a conclusion, that randomly selecting hyperparameters has no sense. Therefore we performed a grid search in search of the best set. Hyperparameters searched were starting value of learing rate and scheduler's patience.

We focused mainly on validation loss and accuracy when choosing final model.



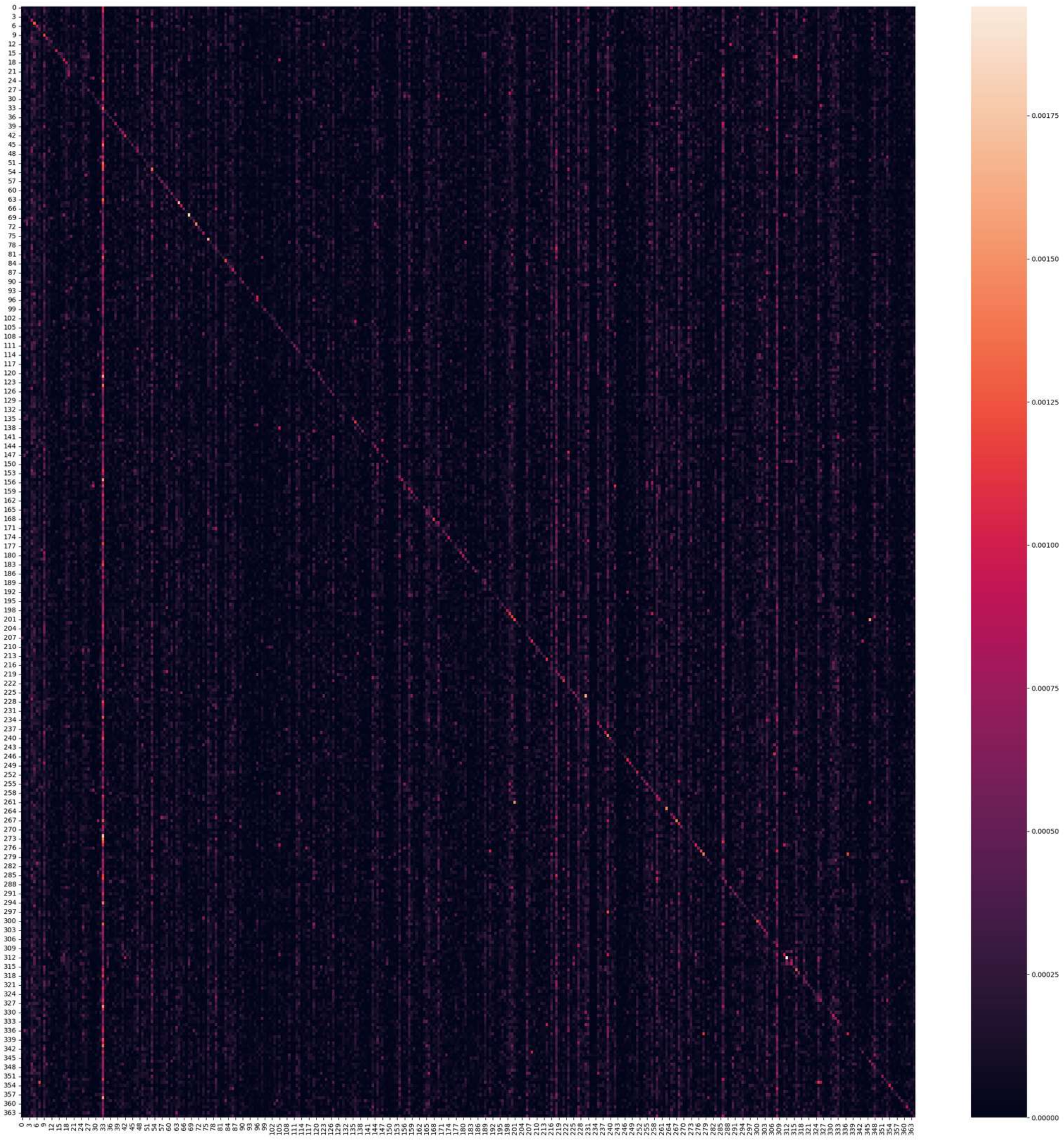


Training with learning rate equal to 0.003 and patience 1 (sandy-tree-20) yields the best results.

Evaluation

We performed some simple evaluation of our final model.

It reached 38% accuracy



Created w

https://wa
Vmlldzoxl

Confusion matrix of our model predictions on test set

As seen on confusion matrix, our model correctly classifies most of the scenarios. It is easily verified as the diagonal is sparking significantly. Some columns stand out, for example class 33 which is 'balcony_exterior'. If one takes look into the data, such a confusion is understandable as images of that class rarely consist only of the balcony itself.