Assignment 3: Image classification on classifysketch dataset

CALLARD Baptiste
ENS Paris-Saclay (Master MVA)
baptiste.callard@ens-paris-saclay.fr

Abstract

The aim of this work is to develop a model capable of classifying the images of the dataset classifysketch with the best accuracy. It is made up of 250 classes of sketches. We will begin by examining the dataset, then discuss the model selection, data augmentation and model tuning that enabled me to achieve 84.7% accuracy on the test dataset using results from [1] and [2] and a ***new data augmentation****.*

1. Introduction

The dataset classifysketch contains grayscale sketch images with a size of 1111×1111 with 250 classes. The training set contains 12,000 images, i.e. 48 images per class. The validation dataset contains 2250 images, 9 images per class.

2. Model selection

First of all, I chose a model that performed well on the data without pre-processing and without using too many resources to train it on a Tesla T4 GPU from Google Colab. I found it instructive to improve the performance of small models than to test all the big models. To keep track of my experiments, I added weights and biases. Then I tested the models using Torchvision and Hugging Face by focusing on the models developed by Timm. I use pre-trained checkpoint on ImageNet with a resize of 224 and dropout of 0.2 for the fully connected. I tested: resnet50, resnet101, resnext50_32x4d, wide_resnet50_2, vit_small_patch16_224.

The little vit_patch16_224 (30.1M Params) was the most promising model using ImageNet's pre-entrained weights (see. models) by looking at accuracy. I have also used the resnet50 as a baseline for the approaches described in the next section.

3. Improving performance

Initially, I compared normalisation to standardisation and obtained better results with normalisation. I then carried out a classic data augmentation: translation, rotation, scaling, shearing, random horizontal flipping. This increased performance to 79.5% on the test set. I tested increasing the size of the model head and freezing the base model for warm-up or not. This resulted in a drop in performance in both cases. I then tested several learning rates, different schedulers and optimizer. I obtained very fast convergence and 80.7% on the test set using a learning rate of 5e-3 and an exponential scheduler with SGD optimizer.

Inspired by the papers from Google Research, Brain Team [1] and [2]. I implemented from scratch Cut-Mix, Mix-up, used SGD, cosine learning rate schedule with and without linear warmup and got 81.7%. I also create a brand new specific data augmentation by expanding randomly the lines thickness using sklearn and got 82.3%. I could again increase performance just by using a bigger model vit-base-patch16-224 and reach 83.6% on test dataset. I finally got on the test set 84.7% by adding the validation set to the the train set during training.

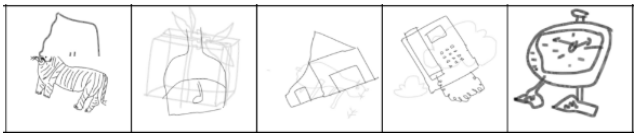

Figure 1. *Cut-Mix : image 1 — Mix-Up : images 2, 3, 4 — Lines thickness augmentation: image 5*

The best configuration is as follows: batch=32, lr=5e-3, optimizer SGD with momentum=0.5, cosine lr schedule without linear warmup, resize=224, with normalisation and using rotation, translation, scale, shearing, random horizontal flip, Cut-mix, Mix-up (Pars. fixed using [2]) and random line thickness augmentations.

4. Conclusion

By proceeding step by step and testing the various tools I had seen in class: CNN, VIT, data augmentation, regularisation, optimiser and scheduler and by using research papers, I obtained a score of 84.7% on the test dataset. Finally, for a bigger project it would be possible to try ensemble learning and to do other data augmentation (see. pytorch doc. for Resnet50 v2) to improve accuracy.

References

[1] Alexander Kolesnikov Dirk Weissenborn Xiaohua Zhai Thomas Unterthiner Mostafa Dehghani Matthias Minderer Georg Heigold Sylvain Gelly Jakob Uszkoreit Neil Houlsby Alexey Dosovitskiy, Lucas Beyer. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. Google Research, Brain Team. 1

[2] Xiaohua Zhai Ross Wightman Jakob Uszkoreit Lucas Beyer Andreas Steiner, Alexander Kolesnikov. How to train your vit? data, augmentation, and regularization in vision transformers, 2021. Google Research, Brain Team. 1