Sketch images classification: Challenge results

Akedjou Achraff ADJILEYE

November 28, 2023

Abstract

In this report, I present my approach and my result for the sketch images classification Kaggle challenge in which I finished ranked ... as part of the Master MVA lecturer: Object Recognition and Computer Vision.

1 Introduction

In this challenge, we have given a dataset containing classes equi-distributed training, validation and test sets. The data are sketches of images from 250 different classes, and the goal is to train a model that achieves the highest accuracy on the test data.

2 Model architecture

For this challenge, we employed an ensemble method involving 5 experts. All experts are Vision Transformers of different sizes, each fine-tuned in an individual way by making different image transformations. They are previously pre-trained on ImageNet 21k and fine-tuned on ImageNet 1K classification task. For the 5 experts we used the google/vit-base-patch16-224 checkpoint available on the Hugging Face hub.

2.1 Experts Agregation

Each expert predicts a class for a given image, resulting in 5 predicted classes per image. The model's predicted class is then determined by a majority vote from the 5 predictions. In case of a tie (such as 2-2-1), the final class is the one predicted by the group of experts with the highest average score. Note that the score of an expert is his achieved accuracy on the validation set.

3 Experiments

Table 1 shows the parameters used to train each experts and the transformation applied on the data:

n°	nhl	size	Data Transformations	accuracy
1	13	93	RRC, RHF, RA, RP	81.289
2	13	93	Id, RR	80.844
3	14	100	Id, RRC, RHF, RA, RP, RR	80.933
4	12	86	Id	79.426
5	13	93	RHF, RVF, RR	77.60

Table 1: Experts specifications, R:Random, RR: Random Rotation, H:Horizontal, V: Vertical, F: Flip, C: Crop, A: Affine, P: Perspective, Id: Identity, nhl: number of hidden layers; size are in millions of parameters

All experts training was started with a batch size of 64 (except 128 the third expert because it's was trained on kaggle notebook on which we have 2T4 GPUs for free than only one for colab), a linear learning rate scheduler with a maximum learning rate of 1e-4 , 0.1 warmup ratio, and a weight decay of 0.02, for 50 epochs. We use the hugging face **Trainer** API for all these experiments. The training stopping procedure was done manually depending on each expert, but in all cases, the goal has always been to maximize accuracy on the validation data.

4 Results and Discussion

We start this challenge with the idea of finetuning the better possible google/vit-base-patch16-224 vision transformer on the data. After several attempts to calibrate the model's hyperparameters and training parameters, we wasn't able to achieve better than 79.426 as accuracy on validation data (expert 4, tab 1).

We then perform a qualitative analysis of the model's errors by visualizing some of expert 1 mistakes (see Figure 1). This analysis prompted us to augment the training data by applying transformations to the images to boost the model's performance in generalization; and led to the training of first expert. Subsequently, aiming to leverage both trained models and further explore image augmentation techniques, we trained 3 other models, varying the image augmentation methods each time, resulting in the development of 3 additional experts.

Ensemble model is a well-known technique in machine learning that enhances model generalization performance (e.g., Random Forest). By combining the predictions of the 5 experts (method described in section 2.1), we achieved a validation accuracy of 83.511, more than 2 points higher than the best expert.

Note that a random chosen transformation in the list of each expert's data transformation methods (see tab 1) is applied to each training image at each forward pass.

5 Submission on Kaggle

The 5 experts agregation used to predict classes on test data gives an accuracy of 84.347.

To take profit on the available validation data, we retrain all experts for 12 epochs (again with hand-crafted stopping method) with a lower learning rate (5e-6), 64 batch size, 0.1 warmup ratio on the train and validation data merged together. We then used the 5 retrained models to make another prediciton on the test data and get 84.637 on the public learderboard.

Figure 1: Some misclassified images by expert 4, for title: bad predicted label/true label

6 Pain points and Conclusion

This challenge was very interesting and fun. The most pain point was about computationnal ressources: colab allows continuous training only for 12 hours and then you must wait 12 hours to have access to a GPU again. We were obliged to switch between colab and kaggle notebook, which is time consuming and definitely not "checkpoint folders saving and managing" friendly.