Kaggle Data Challenge : Sketch Classification

Farah Ellouze
ENS Paris Saclay
farah.ellouze@ens-paris-saclay.fr

Abstract

The report describes the architecture of the model implemented to solve the Kaggle Data Challenge that aims at classifying sketches. First, we will provide a general description of the datastet and the data augmentation methods used. Then we will tackle the model's architecture. Finally, the results will be summed up in the last part.

1. Dataset

1.1. General Description

The dataset contains 250 different classes of sketches adapted from the classifysketch dataset that was divided into training, validation and testing sets. The dataset was difficult to handle for different reasons : sketches can be very tricky to work with since there are multiple ways to represent an object so the intra-classes variance is high. Also, they are freehand sketches so there is always noise around straight lines and some of them were just of low quality.

1.2. Data Augmentation

As the variance intra-classes was very high, the data augmentation methods were simple and did not affect the edges of the objects (elastic deformations) so they do not increase the variance and the model can still be better generalized. We created a pipeline that performs resizing, normalizing, horizontal flip, random rotation, and random translation. The random translation function was adapted from the article [2]. The function implemented in the article was used because it ensures that the image does not get cutout when translated by adding padding. Also, the cutout method was discarded because some classes were wrongly classified as part of other classes (for example, a foot was misclassified as person walking).

2. Model Architecture

The images in the dataset do not contain color or texture, thus the chosen model was not very deep (24 layers). The model implemented had a **GoogleNet** backbone which was pre-trained on Imagenet dataset. Two fully connected layers were added at the end to be trained only on the classifysketch dataset and thus fine-tune the model on the target dataset[1]. Since there are dissimilarities between image classification and sketch classification, even the pretrained layers were retrained on the dataset but with a lower learning rate (0.001 for the backbone layers versus 0.01 for the fully connected layers). The initial strategy was to freeze some the layers of the backbone but it resulted in very poor performances on both the train and validation sets in this case.
Two dropouts layers with probabilities 0.5 and 0.2 respectively were added to the model to avoid overfitting and a relu activation. The used optimizer relied on the Stochastic Gradient Descent method, and, a CosineAnnealingLR (learning scheduler) was implemented to regularize the learning rates and avoid overfitting.

3. Results

The model was trained during 15 epochs but we kept the one of the 13-th epoch because the model was just overfitting the train set afterwards without significant improvement on the validation set. It achieved the following results : Train-set accuracy : 65%, validation-set accuracy : 68%, average validation-set loss : 0.0253, and available score on 30 % of the test set : 70.26%.
The results could have been improved with using a model stacking approach. But this method was not implemented due to a shortage of computation resources.

References

[1] Fevziye İrem Eyiokur, Doğucan Yaman, and Hazım Kemal Ekenel. Sketch classification with deep learning models. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4, 2018. 1

[2] Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient GAN training. *CoRR*, abs/2006.10738, 2020. 1