Deep Learning: Homework 2

Instructed by *Yi Wu*

Due on Mar 24, 2021

Runlong Zhou YaoClass 82 2018011309

Model Details

The core part of my model is adapted from VGG16. My model takes a 3-channel 96×96 picture as input and outputs a 10-dim vector as classification result.

The model consists 5 parts, transforming feature sizes as: $96 \times 96 \times 3 \to 48 \times 48 \times 64 \to 24 \times 24 \times 128 \to 12 \times 12 \times 256 \to 6 \times 6 \times 512 \to 3 \times 3 \times 512$. In each part, there are two to three groups of (conv2d, batchnorm2d, relu), followed by a final maxpool2d whose stride is 2 to make side length halved.

The final output is calculated by a full-connection with $3 \times 3 \times 512$ inputs and 10 outputs.

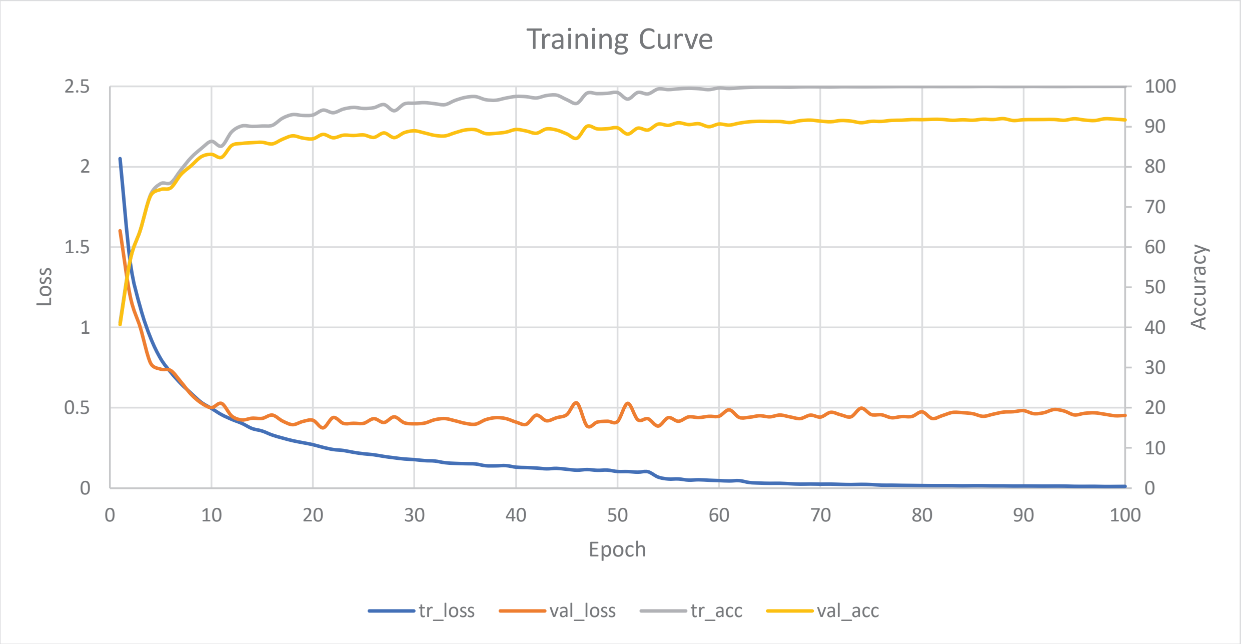
Hyper-Parameters

1. num-epoch: default 100.
2. lr: default 5e-3.
3. weight-decay: default 0.
4. batch-size: default 16.
5. aug-prob: probability to do augmentation, default 0.5.
6. ensemble: usage specified in the tricks, default=6.

Tricks

1. **Data augmentation:** Pepper-noise (deprecated), Scale-then-randomcrop (substituted by ensemble), Random-horizontal-flip, Random-vertical-flip (deprecated), Random-rotation (deprecated).
2. **Ensemble:** Use several (number specified by –ensemble) 96×96 crops of the original 128×128 picture as input, and average softmax(outputs) as ensemble probability. To create the same loss function, first apply log() then use NLLLoss().

Training Curve



Best model is at epoch 88.

Modifications to evaluation.py

1. Line 5: Net → *.
2. Line 35: map_location=device, since my model is not trained on CPU or GPU0.