50.039 – Theory and Practice of Deep Learning

Alex

Week 04: Fine Tuning of neural networks

1 In class Task + homework

- check the AMI that I shared with your amazon account: it is oregon zone, search for owner: 277133844599. You should see an AMI.
- take the 102 class flowers dataset and write a dataset class which can work with a train/val/test split for it.
- take any deep network you like which has pretrained weights (a resnet18 or a mobilenet are training fast, keep your AWS money for the projects rather than using super slow architectures NASNets or VGG)
- train a deep neural network in three different modes:

  A once without loading weights and training all layers.

  B once with loading model weights before training and training all layers,

  C once with loading model weights before training and training only the last two trainable layers (note: for quite some problems, the approach B is better than C)

  For each of these 2 modes select the best epoch by the performance of the model on the validation set. Typically less than 10 epochs should suffice for training when using finetuning. You can run also optionally a selection over a few learning rates. If you use amazon AWS please do not use more than 6 GPU hours. You will need them later for 2 projects, training a GAN and other stuffs

- what loss to choose for a 102-class multiclass dataset?

  what do you need to do for steps when you start with a code like the MNIST training code?

  - write a new dataloader for your training dataset
  - adjust paths for data (and if necessary for label paths/files or files determining splits into train/val/test)
  - decide on some at least basic data augmentation (how to load the images into a fixed size: resizing+some cropping, do more at training time? do what at test time ?)
  - use some deep learning model from the model zoo, load its weights before training
  - think of what results to report for homework submission. a naked code will not do it!

- A note: Calling a model constructor with `pretrained=True` does not tell you what really goes on when one. Check `https://github.com/pytorch/vision/blob/master/torchvision/models/resnet.py` to see what routine is used to load a model.

- for the homework report at least the following:

  - for each of the 3 settings curves of training loss, validation loss and validation accuracy as a function of epochs (for the best setting you found)
  - for each of the 3 settings the test accuracy of the best model
  - observe differences between the validation and the test accuracy of these models