50.021 Artificial Intelligence
Coding Part of Homework 3

Due: every Tuesday, 6PM

Write your own Dataset subclass over 250 or 500 imagenet val images which rescales images to 280 on the smaller side and does five crop without any precoded five-crop transform class but you can use `transforms.ToTensor()`, the pixel normalization and the resizing to the shorter side from toolboxes. To make it clear: For this homework you are not allowed to use pytorch's five crop.

The dataset is `imagenet2500.tar` in: `https://www.dropbox.com/sh/1xii0pcowizw0y3/AAB8srrbYqYNsuNQNRFY` `dl=0` The labels, and helper code (see l8 lecture) are given in codeforstudents subfolder of that lecture.

Test it with a deep neural network on image where smallest side of an image is resized to 280 and you use 224 crops. Let it run in a batchsize of larger than 1. Compare the test performance on your five crop against using only center crop on images resized to 224 on their smallest side.

There are at least two different solution strategies to make this work:

Way 1: Your dataloader should output $datasize * 5$ many samples, with the intention to decompose your `idx` in your dataset class into a data sample index and a crop index. This layer is NOT required to work with shuffling, use it without shuffling. Then adjust your batchsize to five times your desired batchsize, then average predictions over the correct blocks of samples which are the five crops of the same image.

pytorch: You can use the way to read the image by e.g. PIL, run all the pytorch transforms including totensor, then create crops on that object same as you would play on a numpy array. mxnet: you can use the other image transforms first, then use ToTensor() to obtain an `mx.ndarray` object, then create crops on that object same as you would play on a numpy array.

Way 2: You output for one image a 4-dim tensor in your dataset class $(5, c, h, w)$ instead of a 3-dim tensor (c, h, w). The dataloader will then turn it into 5-tensor $(batchsize, 5, c, h, w)$ which you can then decompose when you run a for loop over outputs of your dataloader class.