数据增强学习

Pytorch自带的transforms

Transforms模型位于torchvision中，其调用采用如下方法：

import torchvision.transforms as transforms

transforms.Compose() 方法接收一个 transforms 方法的list为参数，将这些操作组合到一起，返回一个新的tranforms。通常用于包装一个完整的变换操作的pipeline：

1. myTransforms = transforms.Compose([
2. transforms.CenterCrop(10),
3. transforms.ToTensor()
4. ])

本节所提到的所有的transforms的操作都可以以三种数据中的一种为参数：

* PIL格式的Image
* tensor表示的图片，其形状为 (Channel, Height, Weight)
* 一个tensor数组表示的多个图片，其形状为 (Batch, Channel, Height, Weight)

transforms各个函数

 torchvision.transforms.CenterCrop(size)：size为一个整数或者tuple，从图片中心裁剪出size大小的图片

 torchvision.transforms.ColorJitter(brightness=0, contrast=0, saturation=0, hue=0)：对图像的亮度、对比度、饱和度和色调进行调节，每个参数接收一个float值或者一个float tuple (min, max)



 torchvision.transforms.FiveCrop(size)：size为一个整数或者tuple，从图片中心和四个角裁剪出一共五张size大小的图片

 torchvision.transforms.Grayscale(num\_output\_channels=1)：将图片转换为灰度图，参数可取1或3，取3时，三个通道返回同样的值

 torchvision.transforms.Pad(padding, fill=0, padding\_mode='constant')：对图片做padding操作

 torchvision.transforms.RandomAffine(degrees, translate=None, scale=None, shear=None, resample=0, fillcolor=0)：对图像进行随机的仿射变换



 torchvision.transforms.RandomApply(transforms, p=0.5)：接收一个transforms的list为参数，依据它对图片进行随机的变换

 torchvision.transforms.RandomCrop(size, padding=None, pad\_if\_needed=False, fill=0, padding\_mode='constant')：在图片的随机位置剪裁并返回新图像

 torchvision.transforms.RandomGrayscale(p=0.1)：以p的概率将图片随机转换为灰度图

 torchvision.transforms.RandomHorizontalFlip(p=0.5)：以p的概率随机将图片水平翻转

 torchvision.transforms.RandomPerspective(distortion\_scale=0.5, p=0.5, interpolation=2, fill=0)：以p的概率将图片进行随机透视变换



 torchvision.transforms.RandomResizedCrop(size, scale=(0.08, 1.0), ratio=(0.75, 1.3333333333333333), interpolation=2)：将图片随机地进行剪裁得到给定大小或者长宽比的图片

 torchvision.transforms.RandomRotation(degrees, resample=False, expand=False, center=None, fill=None)：随机对图片旋转一个角度

 torchvision.transforms.RandomVerticalFlip(p=0.5)：以p的概率随机将图片垂直（上下）翻转

 torchvision.transforms.Resize(size, interpolation=2)：将图片大小放缩到给定大小

 torchvision.transforms.TenCrop(size, vertical\_flip=False)：将图片的四个角和中心进行裁剪后，返回他们的反转后的图片，默认水平反转

 torchvision.transforms.GaussianBlur(kernel\_size, sigma=(0.1, 2.0))：对图片进行高斯模糊

 torchvision.transforms.RandomInvert([p])：以给定的概率随机反转给定图像的颜色

 torchvision.transforms.RandomPosterize(bits[, p])：通过减少每个颜色通道的比特数，以给定的概率随机地对图像进行分离 bits∈[1-8]

 torchvision.transforms. RandomSolarize(threshold[, p])： 通过反转所有高于阈值的像素值，以给定的概率随机地对图像进行过曝



 torchvision.transforms. RandomAdjustSharpness(sharpness\_factor[, p]) ：以给定的概率随机调整图像的锐度



 torchvision.transforms. RandomAutocontrast([p])：按照指定概率随机对图像进行自动对比度调整（调整当前图像的明暗对比）

 torchvision.transforms. RandomEqualize([p])：用给定的概率随机地对给定图像进行直方图均衡



仅对PIL Image起作用的操作

 torchvision.transforms.RandomChoice(transforms)：接收一个transforms的list为参数，从list中随机挑选一个对图片进行变换

 torchvision.transforms.RandomOrder(transforms)：接收一个transforms的list为参数，将list中的所有变换操作进行随机排序后对图片进行变换

仅对Tensor起作用的操作

 torchvision.transforms.LinearTransformation(transformation\_matrix, mean\_vector)：对tensor进行矩阵变换后减去一个向量

 torchvision.transforms.Normalize(mean, std, inplace=False)：对图片进行标准化

 torchvision.transforms.RandomErasing(p=0.5, scale=(0.02, 0.33), ratio=(0.3, 3.3), value=0, inplace=False)：随机选取图片中的矩形区域并删除其像素值

 torchvision.transforms.ConvertImageDtype(dtype: torch.dtype)：将tensor的数据类型转换为给定的torch.dtype的类型

用于进行转换的操作

 torchvision.transforms.ToPILImage(mode=None)：将一个tensor格式的图片转换为PIL Image格式的图片，不支持torchscript

 torchvision.transforms.ToTensor：将一个PIL Image格式的图片转换为tensor格式的图片，不支持torchscript，**这个很重要**！

timm库的数据增强

1. **from** timm.data **import** Mixup
2. **from** timm.data.random\_erasing **import** RandomErasing
3. **from** timm.data.auto\_augment **import** auto\_augment\_transform
4. **from** timm.data.transforms **import** RandomResizedCropAndInterpolation

 Mixup:将随机的两张样本按比例混合，分类的结果按比例分配；

 Cutout:随机的将样本中的部分区域cut掉，并且填充0像素值，分类的结果不变；

 CutMix:就是将一部分区域cut掉但不填充0像素而是随机填充训练集中的其他数据的区域像素值，分类结果按一定的比例分配；

 AutoAugment:在强化学习(RL)的帮助下学习给定数据集的最佳数据增强策略；

 Random Resized Crop And Interpolation:随机缩放裁剪和插值。

Mixup(mixup\_alpha=1., cutmix\_alpha=0., cutmix\_minmax=None, prob=1.0, switch\_prob=0.5, mode='batch', correct\_lam=True, label\_smoothing=0.1, num\_classes=1000)。 该函数实现了Mixup和CutMix。

mixup\_alpha (float): mixup alpha value, mixup is active if > 0.

cutmix\_alpha (float): cutmix alpha value, cutmix is active if > 0.

cutmix\_minmax (List[float]): cutmix min/max image ratio, cutmix is active and uses this vs alpha if not None.

prob (float): probability of applying mixup or cutmix per batch or element.

switch\_prob (float): probability of switching to cutmix instead of mixup when both are active.

mode (str): how to apply mixup/cutmix params (per 'batch', 'pair' (pair of elements), 'elem' (element).

correct\_lam (bool): apply lambda correction when cutmix bbox clipped by image borders.

label\_smoothing (float): apply label smoothing to the mixed target tensor.

num\_classes (int): number of classes for target.

RandomErasing(probability=0.5, min\_area=0.02, max\_area=1/3, min\_aspect=0.3, max\_aspect=None, mode='const', min\_count=1, max\_count=None, num\_splits=0, device='cuda')。该函数实现了随机擦除，即Cutout。

probability: Probability that the Random Erasing operation will be performed.

min\_area: Minimum percentage of erased area wrt input image area.

max\_area: Maximum percentage of erased area wrt input image area.

min\_aspect: Minimum aspect ratio of erased area. mode: pixel color mode, one of 'const', 'rand', or 'pixel'.

'const' - erase block is constant color of 0 for all channels.

'rand' - erase block is same per-channel random (normal) color

'pixel' - erase block is per-pixel random (normal) color

max\_count: maximum number of erasing blocks per image, area per box is scaled by count.

per-image count is randomly chosen between 1 and this value.

仅对PIL Image起作用的操作

auto\_augment\_transform(config\_str, hparams)。

config\_str: String defining configuration of auto augmentation. Consists of multiple sections separated by dashes ('-'). The first section defines the AutoAugment policy (one of 'v0', 'v0r', 'original', 'originalr').The remaining sections, not order sepecific determine 'mstd' - float std deviation of magnitude noise applied. Ex 'original-mstd0.5' results in AutoAugment with original policy, magnitude\_std 0.5.

hparams: Other hparams (kwargs) for the AutoAugmentation scheme.

RandomResizedCropAndInterpolation(size, scale=(0.08, 1.0), ratio=(3. / 4., 4. / 3.), interpolation='bilinear')。该函数默认三者全使用。

AugMixDataset

AugMixDataset(dataset, num\_splits=2)。该函数的大概流程就是对原图进行k个数据增强操作，然后使用k个权重进行融合得到aug图像。最终与原图进行一定比率融合得到最终augmix图像。