

Note Information

Author : AbbasXu

Date : 2022-08-28

Title : pytorch基础

Keywords : [#pytorch](#) [#torchvision](#) [#PytorchVideo](#) [#torchtext](#)

torchvision

torchvision.datasets

torchvision.datasets 主要包含了一些我们在计算机视觉中常见的数据集，在 0.10.0 版本的 **torchvision** 下，有以下的数据集：

Names	Names	Names	Names
-----	-----	-----	-----
Caltech	CelebA	CIFAR	Cityscapes
EMNIST	FakeData	Fashion-MNIST	Flickr
ImageNet	Kinetics-400	KITTI	KMNIST
PhotoTour	Places365	QMNIST	SBD
SEMEION	STL10	SVHN	UCF101
VOC	WIDERFace		

torchvision.transforms

一些对图像的数据预处理或者是增强的方法

torchvision.models

PyTorch官方也提供了一些预训练好的模型供我们使用

- **Classification**

Names	Names	Names	Names
-----	-----	-----	-----
AlexNet	VGG	ResNet	SqueezeNet
DenseNet	Inception v3	GoogLeNet	ShuffleNet v2
MobileNetV2	MobileNetV3	ResNext	Wide ResNet
MNASNet	EfficientNet	RegNet	持续更新

- **Semantic Segmentation**
| FCN ResNet50 | FCN ResNet101 | DeepLabV3 ResNet50 | DeepLabV3 ResNet101 |
|-----|-----|-----|-----|
| LR-ASPP MobileNetV3-Large | DeepLabV3 MobileNetV3-Large | 未完待续 |
- **Object Detection, instance Segmentation and Keypoint Detection**
| Faster R-CNN | Mask R-CNN | RetinaNet | SSDlite |
|-----|-----|-----|-----|
| SSD | 未完待续 |
- **Video classification**
| ResNet 3D 18 | ResNet MC 18 | ResNet (2+1) D |
|-----|-----|-----|

torchvision.ops

[官方文档](#)

包括但不限于NMS, RoIAlign (MASK R-CNN中应用的一种方法), RoIPool (Fast R-CNN中用到的一种方法)

torchvision.utils

torchvision.utils 为我们提供了一些可视化的方法, 可以帮助我们将若干张图片拼接在一起、可视化检测和分割的效果。

| draw_bounding_boxes(image, boxes[, labels, ...]) | Draws bounding boxes on given image. |

|-----|-----|

| draw_segmentation_masks(image, masks[, ...]) | Draws segmentation masks on given RGB image. |

| draw_keypoints(image, keypoints[, ...]) | Draws Keypoints on given RGB image. |

| flow_to_image(flow) | Converts a flow to an RGB image. |

| make_grid(tensor[, nrow, padding, ...]) | Make a grid of images. |

| save_image(tensor, fp[, format]) | Save a given Tensor into an image file. |

PyTorchVideo简介

组件:

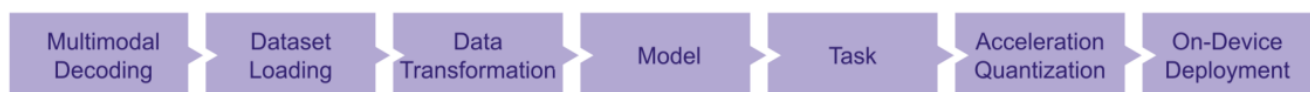


Figure 1: A list of full stack video understanding components provided by PyTorchVideo.

PyTorchVideo的主要部件和亮点

- **基于 PyTorch:**使用 PyTorch 构建。使所有 PyTorch 生态系统组件的使用变得容易。
- **Model Zoo:** PyTorchVideo提供了包含I3D、R(2+1)D、SlowFast、X3D、MViT等SOTA模型的高质量model zoo（目前还在快速扩充中，未来会有更多SOTA model），并且PyTorchVideo的model zoo调用与[PyTorch Hub](#)做了整合，大大简化模型调用，具体的一些调用方法可以参考下面的【使用 PyTorchVideo model zoo】部分。
- **数据预处理和常见数据,** PyTorchVideo支持Kinetics-400, Something-Something V2, Charades, Ava (v2.2), Epic Kitchen, HMDB51, UCF101, Domsev等主流数据集和相应的数据预处理，同时还支持randaug, augmix等数据增强trick。
- **模块化设计:** PyTorchVideo的设计类似于torchvision，也是提供许多模块方便用户调用修改，在PyTorchVideo中具体来说包括data, transforms, layer, model, accelerator等模块，方便用户进行调用和读取。
- **支持多模态:** PyTorchVideo现在对多模态的支持包括了visual和audio，未来会支持更多模态，为多模态模型的发展提供支持。
- **移动端部署优化:** PyTorchVideo支持针对移动端模型的部署优化（使用前述的PyTorchVideo/accelerator模块），模型经过PyTorchVideo优化了最高达**7倍**的提速，并实现了第一个能实时跑在手机端的X3D模型（实验中可以实时跑在2018年的三星Galaxy S8上，具体请见[Android Demo APP](#)）。

使用 PyTorchVideo model zoo

PyTorchVideo提供了三种使用方法，并且给每一种都配备了 **tutorial**

- TorchHub，这些模型都已经在TorchHub存在。我们可以根据实际情况来选择需不需要使用预训练模型。除此之外，官方也给出了TorchHub使用的 [tutorial](#)。
- PySlowFast，使用 [PySlowFast workflow](#) 去训练或测试PyTorchVideo models/datasets.
- [PyTorch Lightning](#)建立一个工作流进行处理，点击查看官方 [tutorial](#)。

torchtext简介

torchtext主要包含了以下的主要组成部分：

- 数据处理工具 torchtext.data.functional、torchtext.data.utils
- 数据集 torchtext.data.datasets
- 词表工具 torchtext.vocab
- 评测指标 torchtext.metrics

构建数据集

- **Field及其使用**
Field是torchtext中定义数据类型以及转换为张量的指令。

```
tokenize = lambda x: x.split()
TEXT = data.Field(sequential=True, tokenize=tokenize, lower=True,
fix_length=200)
LABEL = data.Field(sequential=False, use_vocab=False)
```

其中：

sequential设置数据是否是顺序表示的；

tokenize用于设置将字符串标记为顺序实例的函数

lower设置是否将字符串全部转为小写；

fix_length设置此字段所有实例都将填充到一个固定的长度，方便后续处理；

use_vocab设置是否引入Vocab object，如果为False，则需要保证之后输入field中的data都是numerical的

- **词汇表 (vocab)**
将字符串形式的词语 (word) 转变为数字形式的向量表示 (embedding) 的基本思想是收集一个比较大的语料库 (尽量与所做的任务相关)，在语料库中使用word2vec之类的方法构建词语到向量 (或数字) 的映射关系，之后将这一映射关系应用于当前的任务，将句子中的词语转为向量表示。
- **数据迭代器**
就是torchtext中的DataLoader
- **使用自带数据集**
[官方文档](#)

评测指标 (metric)

并非用准确率来评价。翻译任务常用BLEU (bilingual evaluation understudy) score来评价预测文本和标签文本之间的相似程度。