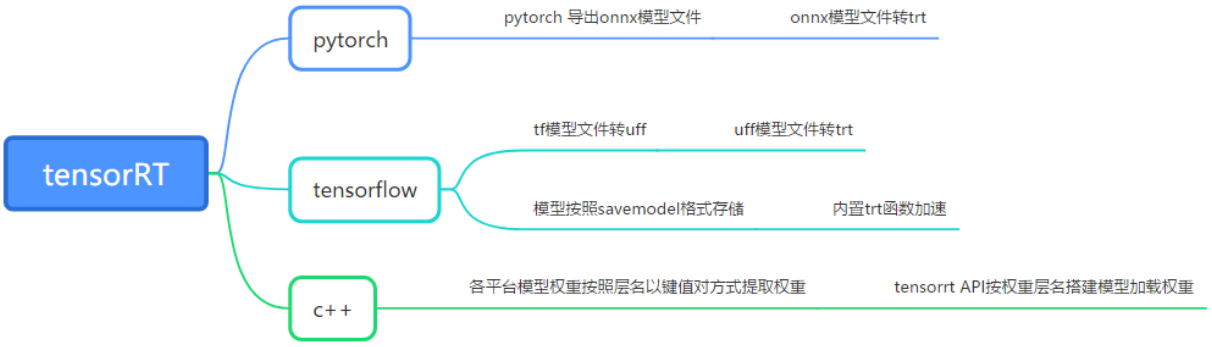
各框架模型tensorrt加速



pytorch2trt

实验环境：

pytorch 1.5.0

tensorrt 7.1

onnx 1.7.0

docker地址：192.168.10.10服务器 镜像 **pytorch2trt** 文件地址 /sample/pytorch2trt

---

1. 模型导出

模型导出时调用转onnx接口转出onnx格式模型结构

```
torch.onnx.export(模型结构，输入大小，输出文件名称，是否打印log)
例:
imgs=torch.randn(1,3,416,416)#模型输入大小
torch.onnx.export(net, imgs,  "yolov2.onnx", verbose=True)
```

2. 模型加速

利用pytorch2trt文件夹下onnx_to_tensorrt.py脚本转换。该脚本是一个通用脚本。

```
python3 onnx_to_tensorrt.py --model yolov2-416
```

同级目录下生成yolov2-416.trt加速模型放入trt_test/yolov2_onnx目录下

3. 模型验证

根据自己模型结构和结果需要编写适合自己模型的验证代码，这里提供一个示例脚本。具体模型加载方式等细节，参考utils目录下yolov2.py代码。

进入trt_test目录下，执行命令

```
python3 trt_yolov2.py --model yolov2-416 --image --filename sample1_10.jpg
```

tensorrt c++ API构建模型加速

实验环境:

pytorch 1.5.0

tensorrt 7.0

onnx 1.7.0

docker地址：192.168.10.10服务器 镜像 **pytorch2trt**

文件地址: /sample/c++

---

1. 模型导出

进入generate_wts文件夹执行命令

```
python3 mobilenet.py
```

会在该脚本路径下生成mobilenet.pth的pytorch模型文件。文件生成之后执行：

```
python3 inference.py
```

会在该脚本路径下生成mobilenet.wts的权重文件。将该权重文件放置mobilenetv2路径下

2. 模型加速

进入mobilenetv2路径下，编译模型转换c++代码，编译完成会生成mobilenet的可执行文件。

```
mkdir build
cd build
cmake ..
make
```

```
./mobilenet -s
```

在build文件夹下生成mobilenet.engine的加速模型文件。

3. 模型验证

```
./mobilenet -d
```

会模拟一个227*227的像素值都为1的图片作为输入并输出模型前向结果。可以将输出值与原模型输出值进行对比（对比代码不提供）。

uff加速

实验环境：

Ubuntu18.04

Cuda10.0+cudnn7.6

tensorflow-gpu 1.14.0

tensorrt7.0

numpy 1.16

192.168.10.10服务器 docker image **tensorflow2trt**

文件地址：/sample/end_to_end_tensorflow_mnist

---

1. 模型导出

tensorflow的pb模型—>uff模型

进入end_to_end_tensorflow_mnist文件夹

```
cd mkdir models
python model.py
```

会在models路径下生成lenet5.pb文件

2.模型转换

```
conver-to-uff models/lenet5.pb
```

此时在models目录下会生成lenet5.uff文件

模型加速验证

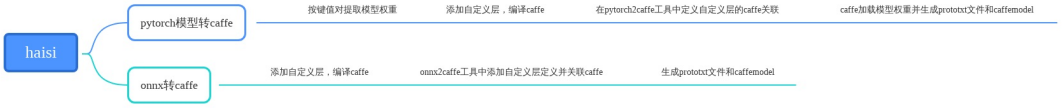
测试加速模型

```
python sample.py -d ./data/
```

tensorflow内置trt转换

目前只有官方提供的例子，该例子用到了tensorflow提供的slim model包去构建模型和加速，该样例封装的过于深，并且跑不通。 官方实例链接

模型转caffe



onnx convert to caffe

实验环境：

numpy 1.15

pytorch 1.5.1

caffe 1.0.0

onnx 1.7

镜像地址：192.168.10.10 镜像 **onnx2caffe**

文件地址：/sample/onnx2caffe-master

1. onnx转caffe

```
cd onnx2caffe
python3 convertCaffe.py ./model/MobileNetV2.onnx ./model/MobileNetV2.prototxt
./model/MobileNetV2.caffemodel
```

生成caffe模型的caffemodel和prototxt

pytorch转caffe

尝试了pytorch2caffe这个工具，但是对于pytorch要求的版本过于低了，导致无法进行测试。可以借鉴解决思路自行定义层与层之间的转换。 该工具github链接