**基于Torchserve模型部署（以mnist为例，测试跑通流程）**

一、拉取torchserve官方镜像

地址<https://hub.docker.com/r/pytorch/torchserve/tags>

担心最新版本不兼容问题，拉取了0.8.0-gpu版本的镜像：docker pull pytorch/torchserve:0.8.0-gpu

二、创建容器、准备文档

基于拉取下来的镜像创建容器：

docker run --rm -it pytorhcserve: 0.8.0-gpu

创建完后自动进入容器，pwd命令查看当前绝对路径:/home/model-server，ls命令查看当前目录文件：“config.properties model-store tmp”，检查后发现该容器只有torchserve的运行环境，并没有官方文档文件，于是在root下通过git拉取文档文件：git clone <https://github.com/pytorch/serve.git>，通过docker命令把serve文档从服务器复制到docker容器中，docker cp /home/piesat/media/ljh/pycharm\_project\_\_ljh/torchserve/ 578363827fa3:/home/model-server。自此model-server目录下多了一个~“torchserve\_examples”文件夹，做到这一步已经准备好了环境和配置文件。

三、开始打包模型，生成mar

如果有需要需要把pth文件先转成ph格式（pthTopt.py）；

在容器内到Cd torchserve\_examples/serve中打包模型：

torch-model-archiver --model-name mnist --version 1.0 --model-file examples/image\_classifier/mnist/mnist.py --serialized-file examples/image\_classifier/mnist/mnist\_cnn.pt --handler examples/image\_classifier/mnist/mnist\_handler.py --extra-files examples/image\_classifier/mnist/mnist\_ts.json

执行完这个语句后会在serve文件夹中生成一个新的文件，文件名为mnist.mar。把mar文件mv到model\_store下。把更改过的容器commit成新镜像：docker commit pytorch/torchserve:1.2。

四、模型部署

基于pytorch/torchserve:1.2镜像创建新的容器并设置端口号

docker run --rm -it -p 8082:8080 -p 8081:8081 pytorhcserve:1.2

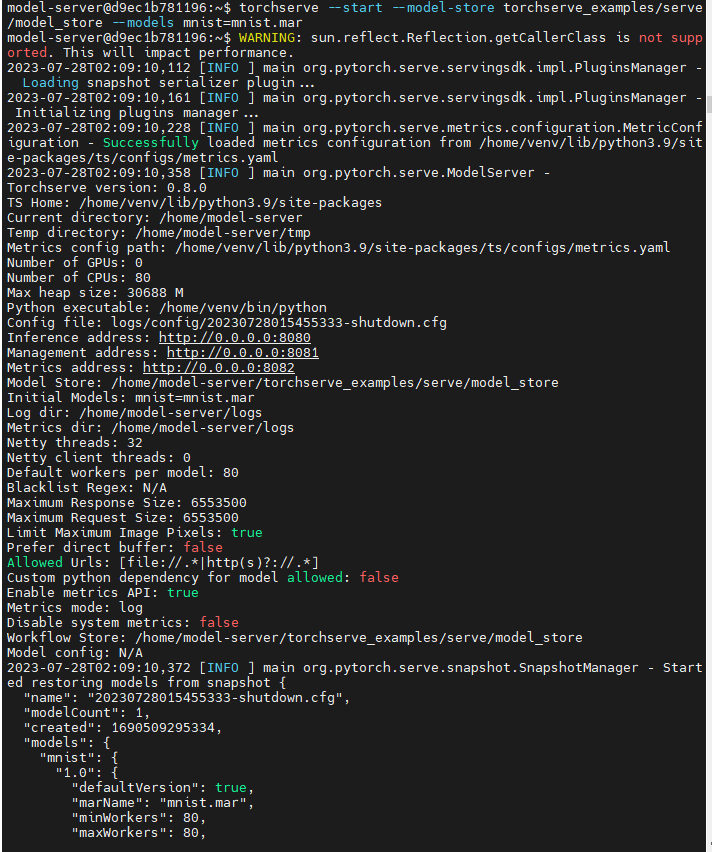
这样设置暴露的7070，7071的端口没有映射，不成功！需要给暴露的端口进行映射，使用如下命令：

docker run --rm -p 8083:8080 -p 8081:8081 -p 8082:8082 -p 7070:7070 -p7071:7071 pytorch/torchserve:1.2

进入容器，在对应目录下启动torchserve服务：

* torchserve --start --model-store torchserve\_examples/serve/model-store --models mnist=mnist.mar
* torchserve --start --model-store cropunet/model-store --models cropunet=cropunet.mar

注意：更换模型前一定要删除外面的logs!!!!!



五、模型推理：

在服务器上root下测试：

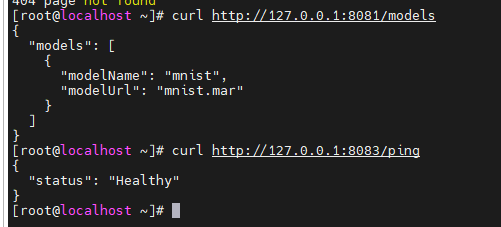
Ping连接状态：curl http://localhost:8083/ping

模型加载情况：curl http://localhost:8081/models

在本地测试（windows）下测试：

Ping连接状态：curl http://192.168.8.27:8083/ping

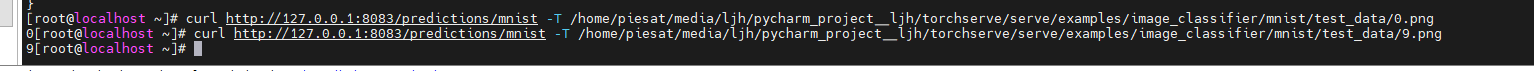
模型加载情况：curl http://192.168.8.27:8081/models



测试图像：

* curl http://127.0.0.1:8083/predictions/cropunet -T /home/piesat/media/ljh/pycharm\_project\_\_ljh/torchserve\_examples/serve/examples/image\_classifier/mnist/test\_data/9.png

9



* curl -X POST http://127.0.0.1:8083/predictions/densenet161 -T /home/piesat/media/ljh/pycharm\_project\_\_ljh/torchserve\_examples/kitten.jpg

curl http://127.0.0.1:8083/predictions/cropunet -T /home/piesat/media/ljh/pycharm\_project\_\_ljh/unet-pytorch/Medical\_Datasets/Images/99\_pz\_0\_2048.png

torch-model-archiver --model-name cropunet --version 1.0 --model-file unet.py --serialized-file Crop\_unet.pt --handler modelhandler.py --extra-files "vgg.py,resnet.py" --export-path model-store -f

torchserve --start --ncs --model-store cropunet/model-store --models cropunet=cropunet.mar

curl -X POST http://127.0.0.1:8083/predictions/cropunet -T /home/piesat/media/ljh/pycharm\_project\_\_ljh/unet-pytorch/cropunet/ 2\_ny\_0\_0.png