Lab 5 - 配置Container进行云上训练或推理

实验目的

1. 理解Container机制
2. 使用Container进行自定义深度学习训练或推理

实验环境

- PyTorch==1.5.0
- Docker Engine

实验原理

计算集群调度管理，与云上训练和推理的基本知识

实验内容

具体步骤

1. 安装最新版Docker Engine，完成实验环境设置
2. 运行一个alpine容器
    1. Pull alpine docker image
    2. 运行docker container，并列出当前目录内容
    3. 使用交互式方式启动docker container，并查看当前目录内容
    4. 退出容器
3. Docker部署PyTorch训练程序，并完成模型训练
    1. 编写Dockerfile：使用含有cuda10.1的基础镜像，编写能够运行MNIST样例的Dockerfile
    2. Build镜像
    3. 使用该镜像启动容器，并完成训练过程
    4. 获取训练结果
4. Docker部署PyTorch推理程序，并完成一个推理服务
    1. 克隆TorchServe源码
    2. 编写基于GPU的TorchServe镜像
    3. 使用TorchServe镜像启动一个容器
    4. 使用TorchServe进行模型推理
    5. 返回推理结果，验证正确性

实验报告

实验环境

硬件环境	CPU（vCPU数目）	Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz
	GPU(型号，数目)	N/A
软件环境	OS版本	Ubuntu 20.04 LTS on VisualBox
	深度学习框架 python包名称及版本	Pytorch 1.5.0 with Python 3.8.5
	CUDA版本	N/A

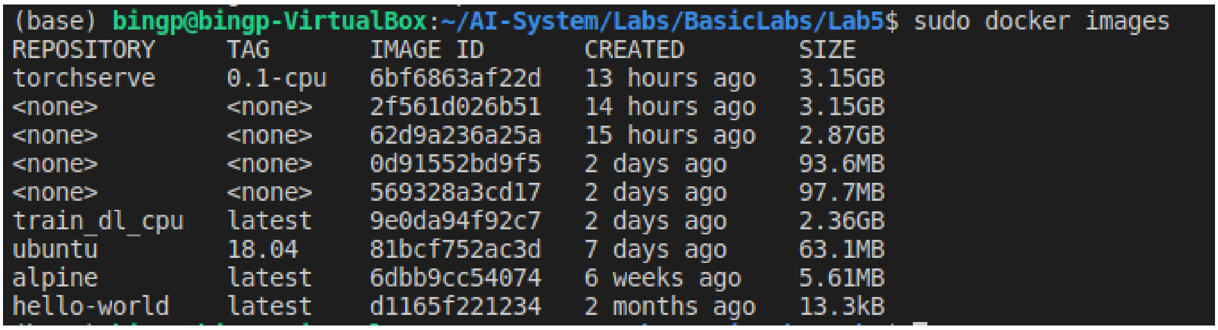
实验结果

1. 使用Docker部署PyTorch MNIST 训练程序，以交互的方式在容器中运行训练程序。提交以下内容：
    1. 创建模型训练镜像，并提交Dockerfile

       由于该镜像需要通过conda下载pytorch，为了加快速度，我修改了一下Dockerfile以加快速度

```
# 继承自哪个基础镜像

FROM ubuntu:18.04

# 创建镜像中的文件夹，用于存储新的代码或文件

RUN mkdir -p /src/app


# WORKDIR指令设置Dockerfile中的任何RUN，CMD，ENTRPOINT，COPY和ADD指令的工作目录

WORKDIR /src/app


# 拷贝本地文件到Docker镜像中相应目录

COPY pytorch_mnist_basic.py /src/app


# 需要安装的依赖



RUN apt-get update && apt-get install wget bzip2 -y

RUN wget https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86_64.sh -O miniconda.sh

RUN bash miniconda.sh -b -p /opt/conda

ENV PATH /opt/conda/bin:$PATH


RUN conda config --set show_channel_urls yes

RUN conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/pytorch/
RUN conda install pytorch torchvision cpuonly -c pytorch



# 容器启动命令

CMD [ "python", "pytorch_mnist_basic.py" ]
```

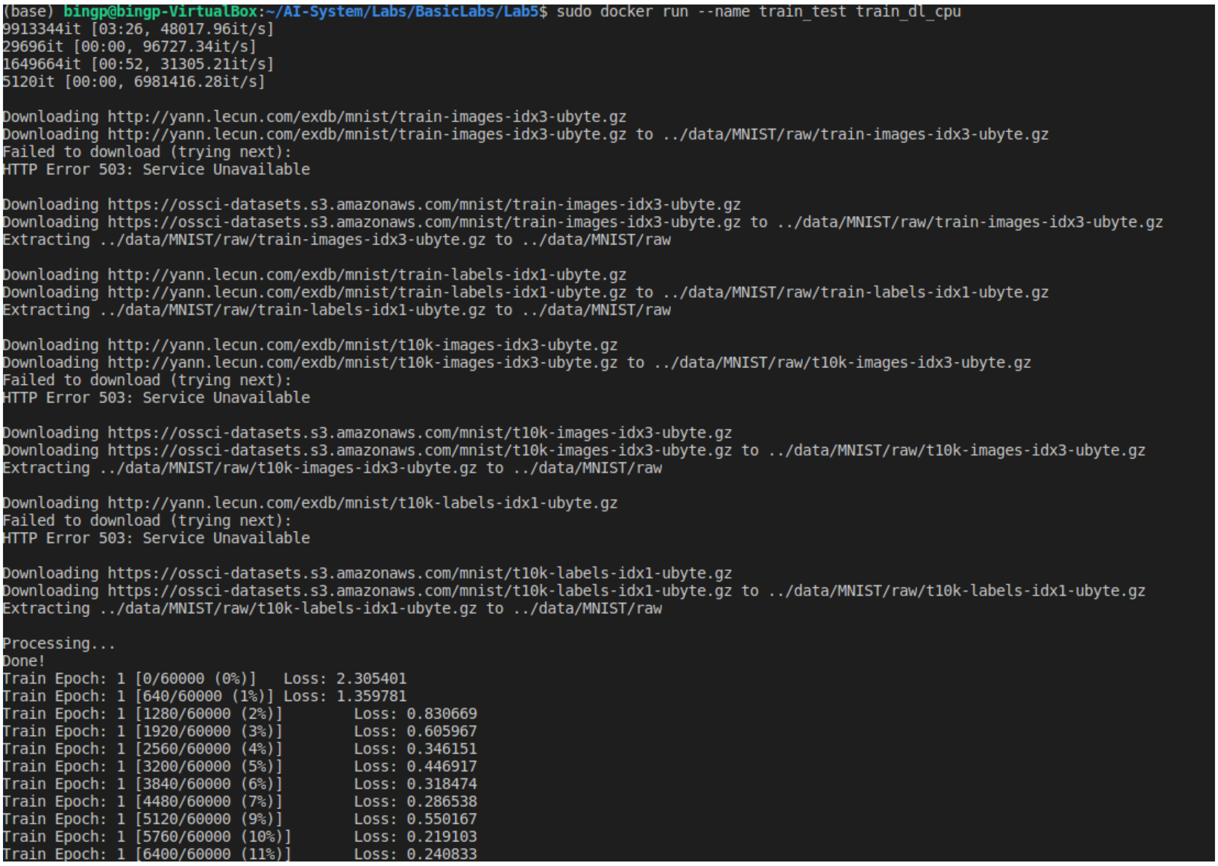
2. 提交镜像构建成功的日志



```
(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ sudo docker images
REPOSITORY     TAG        IMAGE ID       CREATED        SIZE
torchserve     0.1-cpu    6bf6863af22d   13 hours ago   3.15GB
<none>         <none>     2f561d026b51   14 hours ago   3.15GB
<none>         <none>     62d9a236a25a   15 hours ago   2.87GB
<none>         <none>     0d91552bd9f5   2 days ago     93.6MB
<none>         <none>     569328a3cd17   2 days ago     97.7MB
train_dl_cpu   latest     9e0da94f92c7   2 days ago     2.36GB
ubuntu         18.04      81bcf752ac3d   7 days ago     63.1MB
alpine         latest     6dbb9cc54074   6 weeks ago    5.61MB
hello-world    latest     d1165f221234   2 months ago   13.3kB
```

<none>是几次由于网络问题失败的构建

3. 启动训练程序，提交训练成功日志（例如：MNIST训练日志截图）

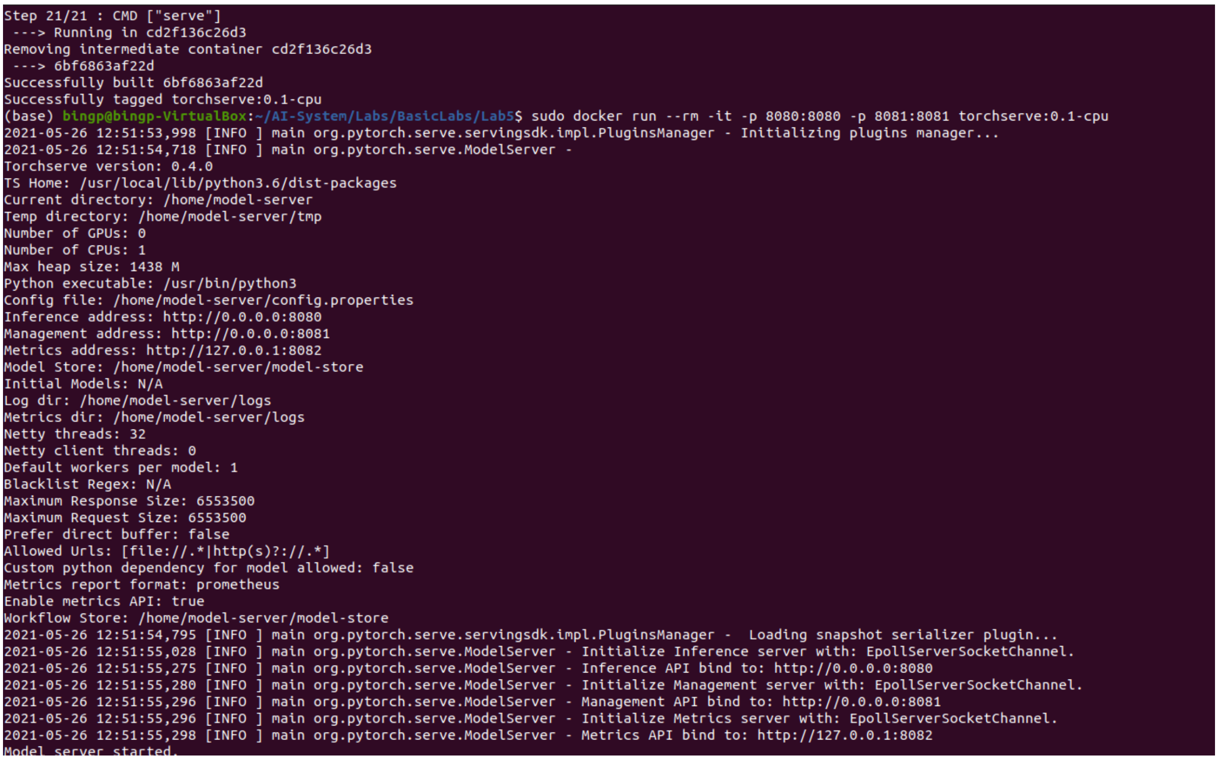


```
(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ sudo docker run --name train_test train_dl_cpu
9913344it [03:26, 48017.96it/s]
29696it [00:00, 96727.34it/s]
1649664it [00:52, 31305.21it/s]
5120it [00:00, 6981416.28it/s]

Downloading http://yann.lecun.com/exdb/mnist/train-images-idx3-ubyte.gz
Downloading http://yann.lecun.com/exdb/mnist/train-images-idx3-ubyte.gz to ../data/MNIST/raw/train-images-idx3-ubyte.gz
Failed to download (trying next):
HTTP Error 503: Service Unavailable

Downloading https://ossci-datasets.s3.amazonaws.com/mnist/train-images-idx3-ubyte.gz
Downloading https://ossci-datasets.s3.amazonaws.com/mnist/train-images-idx3-ubyte.gz to ../data/MNIST/raw/train-images-idx3-ubyte.gz
Extracting ../data/MNIST/raw/train-images-idx3-ubyte.gz to ../data/MNIST/raw

Downloading http://yann.lecun.com/exdb/mnist/train-labels-idx1-ubyte.gz
Downloading http://yann.lecun.com/exdb/mnist/train-labels-idx1-ubyte.gz to ../data/MNIST/raw/train-labels-idx1-ubyte.gz
Extracting ../data/MNIST/raw/train-labels-idx1-ubyte.gz to ../data/MNIST/raw

Downloading http://yann.lecun.com/exdb/mnist/t10k-images-idx3-ubyte.gz
Downloading http://yann.lecun.com/exdb/mnist/t10k-images-idx3-ubyte.gz to ../data/MNIST/raw/t10k-images-idx3-ubyte.gz
Failed to download (trying next):
HTTP Error 503: Service Unavailable

Downloading https://ossci-datasets.s3.amazonaws.com/mnist/t10k-images-idx3-ubyte.gz
Downloading https://ossci-datasets.s3.amazonaws.com/mnist/t10k-images-idx3-ubyte.gz to ../data/MNIST/raw/t10k-images-idx3-ubyte.gz
Extracting ../data/MNIST/raw/t10k-images-idx3-ubyte.gz to ../data/MNIST/raw

Downloading http://yann.lecun.com/exdb/mnist/t10k-labels-idx1-ubyte.gz
Failed to download (trying next):
HTTP Error 503: Service Unavailable

Downloading https://ossci-datasets.s3.amazonaws.com/mnist/t10k-labels-idx1-ubyte.gz
Downloading https://ossci-datasets.s3.amazonaws.com/mnist/t10k-labels-idx1-ubyte.gz to ../data/MNIST/raw/t10k-labels-idx1-ubyte.gz
Extracting ../data/MNIST/raw/t10k-labels-idx1-ubyte.gz to ../data/MNIST/raw

Processing...
Done!
Train Epoch: 1 [0/60000 (0%)]   Loss: 2.305401
Train Epoch: 1 [640/60000 (1%)] Loss: 1.359781
Train Epoch: 1 [1280/60000 (2%)]        Loss: 0.830669
Train Epoch: 1 [1920/60000 (3%)]        Loss: 0.605967
Train Epoch: 1 [2560/60000 (4%)]        Loss: 0.346151
Train Epoch: 1 [3200/60000 (5%)]        Loss: 0.446917
Train Epoch: 1 [3840/60000 (6%)]        Loss: 0.318474
Train Epoch: 1 [4480/60000 (7%)]        Loss: 0.286538
Train Epoch: 1 [5120/60000 (9%)]        Loss: 0.550167
Train Epoch: 1 [5760/60000 (10%)]       Loss: 0.219103
Train Epoch: 1 [6400/60000 (11%)]       Loss: 0.240833
```

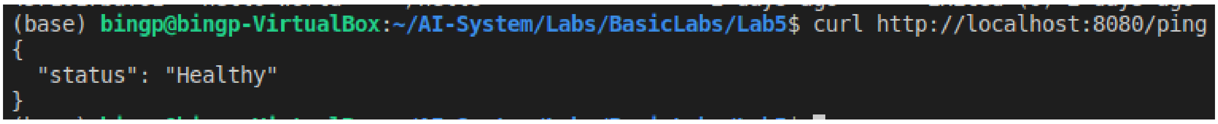
2. 使用Docker部署MNIST模型的推理服务，并进行推理。提交以下内容：

    1. 创建模型推理镜像，并提交Dockerfile

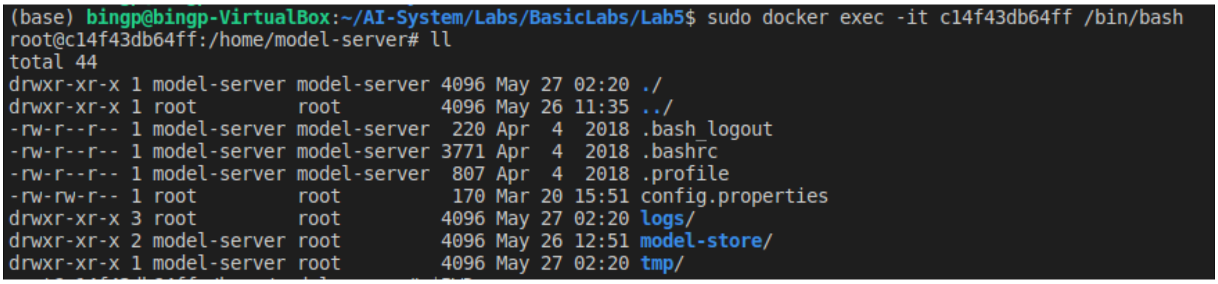
2. 启动容器，访问TorchServe API，提交返回结果日志



```
Step 21/21 : CMD ["serve"]
 ---> Running in cd2f136c26d3
Removing intermediate container cd2f136c26d3
 ---> 6bf6863af22d
Successfully built 6bf6863af22d
Successfully tagged torchserve:0.1-cpu
(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ sudo docker run --rm -it -p 8080:8080 -p 8081:8081 torchserve:0.1-cpu
2021-05-26 12:51:53,998 [INFO ] main org.pytorch.serve.servingsdk.impl.PluginsManager - Initializing plugins manager...
2021-05-26 12:51:54,718 [INFO ] main org.pytorch.serve.ModelServer -
Torchserve version: 0.4.0
TS Home: /usr/local/lib/python3.6/dist-packages
Current directory: /home/model-server
Temp directory: /home/model-server/tmp
Number of GPUs: 0
Number of CPUs: 1
Max heap size: 1438 M
Python executable: /usr/bin/python3
Config file: /home/model-server/config.properties
Inference address: http://0.0.0.0:8080
Management address: http://0.0.0.0:8081
Metrics address: http://127.0.0.1:8082
Model Store: /home/model-server/model-store
Initial Models: N/A
Log dir: /home/model-server/logs
Metrics dir: /home/model-server/logs
Netty threads: 32
Netty client threads: 0
Default workers per model: 1
Blacklist Regex: N/A
Maximum Response Size: 6553500
Maximum Request Size: 6553500
Prefer direct buffer: false
Allowed Urls: [file://.*|http(s)?://.*]
Custom python dependency for model allowed: false
Metrics report format: prometheus
Enable metrics API: true
Workflow Store: /home/model-server/model-store
2021-05-26 12:51:54,795 [INFO ] main org.pytorch.serve.servingsdk.impl.PluginsManager -  Loading snapshot serializer plugin...
2021-05-26 12:51:55,028 [INFO ] main org.pytorch.serve.ModelServer - Initialize Inference server with: EpollServerSocketChannel.
2021-05-26 12:51:55,275 [INFO ] main org.pytorch.serve.ModelServer - Inference API bind to: http://0.0.0.0:8080
2021-05-26 12:51:55,280 [INFO ] main org.pytorch.serve.ModelServer - Initialize Management server with: EpollServerSocketChannel.
2021-05-26 12:51:55,296 [INFO ] main org.pytorch.serve.ModelServer - Management API bind to: http://0.0.0.0:8081
2021-05-26 12:51:55,296 [INFO ] main org.pytorch.serve.ModelServer - Initialize Metrics server with: EpollServerSocketChannel.
2021-05-26 12:51:55,298 [INFO ] main org.pytorch.serve.ModelServer - Metrics API bind to: http://127.0.0.1:8082
Model server started.
```



```
(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ curl http://localhost:8080/ping
{
  "status": "Healthy"
}
```



```
(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ sudo docker exec -it c14f43db64ff /bin/bash
root@c14f43db64ff:/home/model-server# ll
total 44
drwxr-xr-x 1 model-server model-server 4096 May 27 02:20 ./
drwxr-xr-x 1 root         root         4096 May 26 11:35 ../
-rw-r--r-- 1 model-server model-server  220 Apr  4  2018 .bash_logout
-rw-r--r-- 1 model-server model-server 3771 Apr  4  2018 .bashrc
-rw-r--r-- 1 model-server model-server  807 Apr  4  2018 .profile
-rw-rw-r-- 1 root         root          170 Mar 20 15:51 config.properties
drwxr-xr-x 3 root         root         4096 May 27 02:20 logs/
drwxr-xr-x 2 model-server root         4096 May 26 12:51 model-store/
drwxr-xr-x 1 model-server root         4096 May 27 02:20 tmp/
```

3. 使用训练好的模型，启动TorchServe，在新的终端中，使用一张图片进行推理服务。提交图片和推理程序返回结果截图。

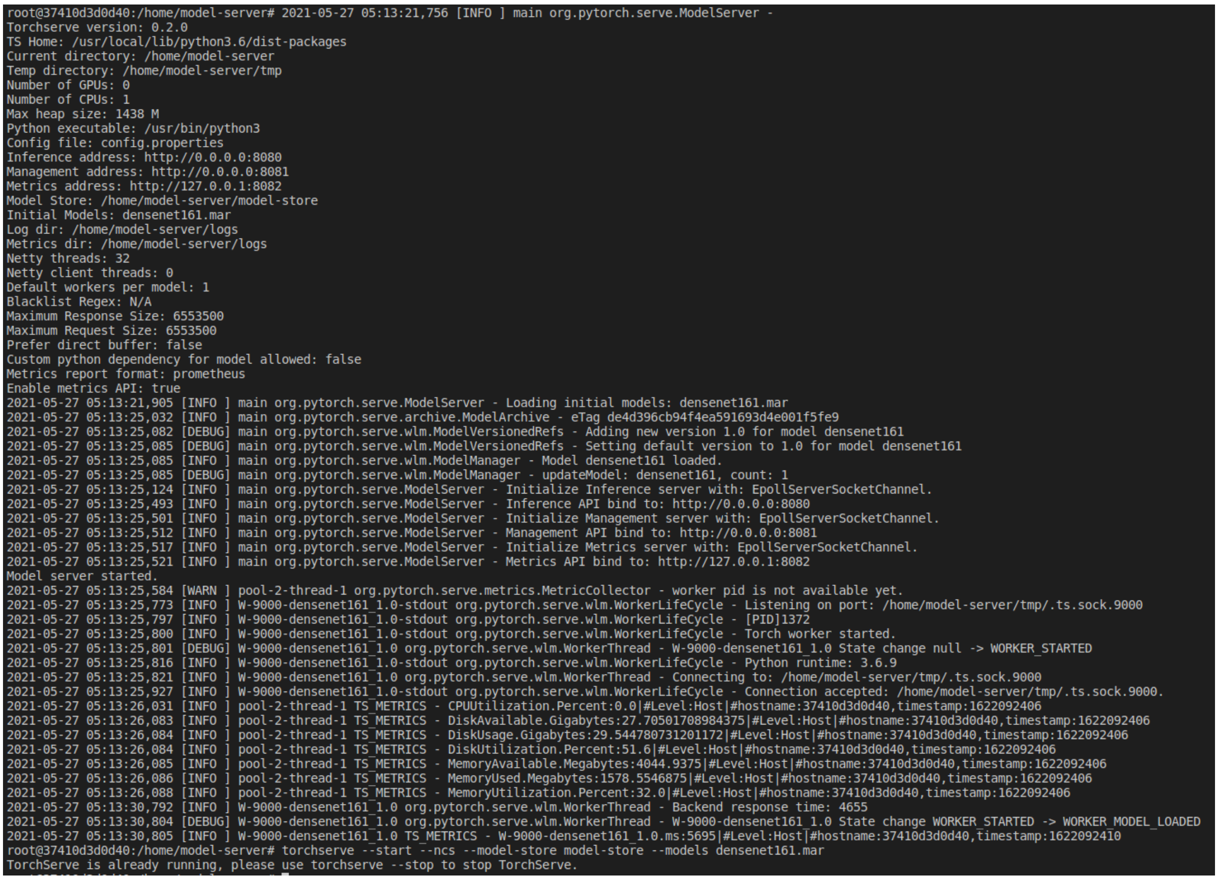
   做到这里，启动TorchServe后一直报错

   ```
   W-9000-densenet161_1.0 org.pytorch.serve.wlm.BatchAggregator - Load model failed: densenet161, error: Worker died.
   ```

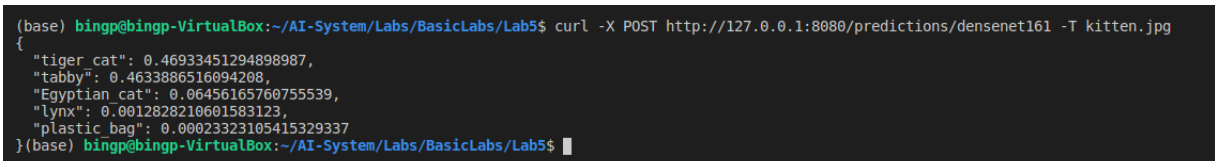
   并且无法完成推理，最终我参照了这个issue No module named 'image_classifier' when following steps given in densenet161 example · Issue #966 · pytorch/serve (github.com) ，使用了github上serve库中图像分类的模型，并重新安装了相应依赖

```
torch-model-archiver --model-name densenet161 --version 1.0 --model-file
/home/image_classifier/densenet_161/model.py --serialized-file /home/model-
server/model-store/densenet161-8d451a50.pth --handler image_classifier --extra-
files /home/image_classifier/index_to_name.json --export-path /home/model-
server/model-store --force
/home/model-server/model-store/densenet161.mar .root@37410d3d0d40:/home# WARNING -
Overwriting /home/model-serve

apt-get install python3 python3-dev python3-pip openjdk-11-jre-headless git wget
curl -y

python3 -m pip install torch torchvision torch-model-archiver torchserve==0.2.0
```

最终serve成功运行并完成了推理



```
root@37410d3d0d40:/home/model-server# 2021-05-27 05:13:21,756 [INFO ] main org.pytorch.serve.ModelServer -
Torchserve version: 0.2.0
TS Home: /usr/local/lib/python3.6/dist-packages
Current directory: /home/model-server
Temp directory: /home/model-server/tmp
Number of GPUs: 0
Number of CPUs: 1
Max heap size: 1438 M
Python executable: /usr/bin/python3
Config file: config.properties
Inference address: http://0.0.0.0:8080
Management address: http://0.0.0.0:8081
Metrics address: http://127.0.0.1:8082
Model Store: /home/model-server/model-store
Initial Models: densenet161.mar
Log dir: /home/model-server/logs
Metrics dir: /home/model-server/logs
Netty threads: 32
Netty client threads: 0
Default workers per model: 1
Blacklist Regex: N/A
Maximum Response Size: 6553500
Maximum Request Size: 6553500
Prefer direct buffer: false
Custom python dependency for model allowed: false
Metrics report format: prometheus
Enable metrics API: true
2021-05-27 05:13:21,905 [INFO ] main org.pytorch.serve.ModelServer - Loading initial models: densenet161.mar
2021-05-27 05:13:25,032 [INFO ] main org.pytorch.serve.archive.ModelArchive - eTag de4d396cb94f4ea591693d4e001f5fe9
2021-05-27 05:13:25,082 [DEBUG] main org.pytorch.serve.wlm.ModelVersionedRefs - Adding new version 1.0 for model densenet161
2021-05-27 05:13:25,085 [DEBUG] main org.pytorch.serve.wlm.ModelVersionedRefs - Setting default version to 1.0 for model densenet161
2021-05-27 05:13:25,085 [INFO ] main org.pytorch.serve.wlm.ModelManager - Model densenet161 loaded.
2021-05-27 05:13:25,085 [DEBUG] main org.pytorch.serve.wlm.ModelManager - updateModel: densenet161, count: 1
2021-05-27 05:13:25,124 [INFO ] main org.pytorch.serve.ModelServer - Initialize Inference server with: EpollServerSocketChannel.
2021-05-27 05:13:25,493 [INFO ] main org.pytorch.serve.ModelServer - Inference API bind to: http://0.0.0.0:8080
2021-05-27 05:13:25,501 [INFO ] main org.pytorch.serve.ModelServer - Initialize Management server with: EpollServerSocketChannel.
2021-05-27 05:13:25,512 [INFO ] main org.pytorch.serve.ModelServer - Management API bind to: http://0.0.0.0:8081
2021-05-27 05:13:25,517 [INFO ] main org.pytorch.serve.ModelServer - Initialize Metrics server with: EpollServerSocketChannel.
2021-05-27 05:13:25,521 [INFO ] main org.pytorch.serve.ModelServer - Metrics API bind to: http://127.0.0.1:8082
Model server started.
2021-05-27 05:13:25,584 [WARN ] pool-2-thread-1 org.pytorch.serve.metrics.MetricCollector - worker pid is not available yet.
2021-05-27 05:13:25,773 [INFO ] W-9000-densenet161_1.0-stdout org.pytorch.serve.wlm.WorkerLifeCycle - Listening on port: /home/model-server/tmp/.ts.sock.9000
2021-05-27 05:13:25,797 [INFO ] W-9000-densenet161_1.0-stdout org.pytorch.serve.wlm.WorkerLifeCycle - [PID]1372
2021-05-27 05:13:25,800 [INFO ] W-9000-densenet161_1.0-stdout org.pytorch.serve.wlm.WorkerLifeCycle - Torch worker started.
2021-05-27 05:13:25,801 [DEBUG] W-9000-densenet161_1.0 org.pytorch.serve.wlm.WorkerThread - W-9000-densenet161_1.0 State change null -> WORKER_STARTED
2021-05-27 05:13:25,816 [INFO ] W-9000-densenet161_1.0-stdout org.pytorch.serve.wlm.WorkerLifeCycle - Python runtime: 3.6.9
2021-05-27 05:13:25,821 [INFO ] W-9000-densenet161_1.0 org.pytorch.serve.wlm.WorkerThread - Connecting to: /home/model-server/tmp/.ts.sock.9000
2021-05-27 05:13:25,927 [INFO ] W-9000-densenet161_1.0-stdout org.pytorch.serve.wlm.WorkerLifeCycle - Connection accepted: /home/model-server/tmp/.ts.sock.9000.
2021-05-27 05:13:26,031 [INFO ] pool-2-thread-1 TS_METRICS - CPUUtilization.Percent:0.0|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:26,083 [INFO ] pool-2-thread-1 TS_METRICS - DiskAvailable.Gigabytes:27.70501708984375|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:26,084 [INFO ] pool-2-thread-1 TS_METRICS - DiskUsage.Gigabytes:29.544780731201172|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:26,084 [INFO ] pool-2-thread-1 TS_METRICS - DiskUtilization.Percent:51.6|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:26,085 [INFO ] pool-2-thread-1 TS_METRICS - MemoryAvailable.Megabytes:4044.9375|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:26,086 [INFO ] pool-2-thread-1 TS_METRICS - MemoryUsed.Megabytes:1578.5546875|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:26,088 [INFO ] pool-2-thread-1 TS_METRICS - MemoryUtilization.Percent:32.0|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092406
2021-05-27 05:13:30,792 [INFO ] W-9000-densenet161_1.0 org.pytorch.serve.wlm.WorkerThread - Backend response time: 4655
2021-05-27 05:13:30,804 [DEBUG] W-9000-densenet161_1.0 org.pytorch.serve.wlm.WorkerThread - W-9000-densenet161_1.0 State change WORKER_STARTED -> WORKER_MODEL_LOADED
2021-05-27 05:13:30,805 [INFO ] W-9000-densenet161_1.0 TS_METRICS - W-9000-densenet161_1.0.ms:5695|#Level:Host|#hostname:37410d3d0d40,timestamp:1622092410
root@37410d3d0d40:/home/model-server# torchserve --start --ncs --model-store model-store --models densenet161.mar
TorchServe is already running, please use torchserve --stop to stop TorchServe.
```



```
(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ curl -X POST http://127.0.0.1:8080/predictions/densenet161 -T kitten.jpg
{
  "tiger_cat": 0.46933451294898987,
  "tabby": 0.4633886516094208,
  "Egyptian_cat": 0.06456165760755539,
  "lynx": 0.0012828210601583123,
  "plastic_bag": 0.00023323105415329337
}(base) bingp@bingp-VirtualBox:~/AI-System/Labs/BasicLabs/Lab5$ 
```

> 如果助教/老师还在维护该项目的话可以加一点说明，这个地方还蛮坑的（

参考代码

本次实验基本教程:

- 1. 实验环境设置
- 2. 运行你的第一个容器 - 内容，步骤，作业
- 3. Docker部署PyTorch训练程序 - 内容，步骤，作业

- 4. Docker部署PyTorch推理程序 - 内容，步骤，作业
- 5. 进阶学习

参考资料

- Docker Tutorials and Labs
- A comprehensive tutorial on getting started with Docker!
- Please-Contain-Yourself
- Create TorchServe docker image