香港中文大學
The Chinese University of Hong Kong

CMSC5743
Lab06 TensorRT

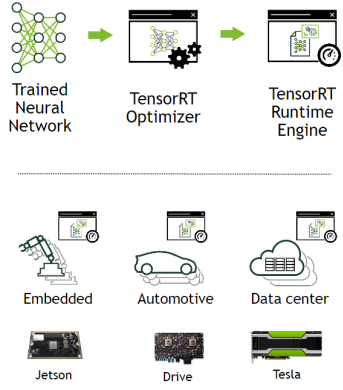
Qi Sun

(Latest update: November 2, 2020)

Fall 2020

TensorRT

- TensorRT official document: https://docs.nvidia.com/deeplearning/tensorrt/index.html.
- A high-performance neural network inference optimizer and runtime engine for production deployment, not for model training.
- A programmable inference accelerator.



TensorRT

Performance Metrics

- Throughput: samples/second or inferences/second.
- Efficiency: amoumt of throughput delivered unit-power.
- Latency: time to execute an inference.
- Accuracy: model accuracy (may degrades due to deployment optimizations: quantization, mixed precision, pruning, interception, *etc.*).
- Memory usage: the host and device memory consumption.

Useful materials

- Forum: https://forums.developer.nvidia.com/c/ai-deep-learning/libraries-sdk/tensorrt/92.
- C++ Library: https://github.com/NVIDIA/TensorRT.

Installation

The most important thing is to read the official guide carefully.

- Some shell scripts are in ./Lab06-code/install-test.sh
- Setup CUDA environment: https://i.cse.cuhk.edu.hk/technical/gpgpu-hpc-service/cuda/.
- Install Python packages: pip install -r requirements.txt
- It's recommended to use virtual environment: python -m virtualenv env
- I use TensorRT 7.2.1 for CentOS/RedHat 7 and CUDA 11.0 TAR package.
- Follow the Tar File Installation to install the TAR package.
- We don't use TensorFlow in this lab, so don't need to install Python UFF wheel file.

```
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/python> pip install tensorrt-7.2.1.6-cp36-none-linux_x86_64.whl
Processing ./tensorrt-7.2.1.6-cp36-none-linux_x86_64.whl
Installing collected packages: tensorrt
Successfully installed tensorrt-7.2.1.6
WARNING: You are using pip version 20.2.3; however, version 20.2.4 is available.
You should consider upgrading via the '/research/d1/qsun/course/cmsc5743-lab06/env/bin/python -m pip install --upgrade pip' command.
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/python>
```

Installation

Check Installation

```
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6> ls
bin  data  doc  graphsurgeon  include  lib  onnx_graphsurgeon  python  samples  targets  TensorRT-Release-Notes.pdf  uff
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6> ls ./lib
libmyelin_compiler_static.a         libmyelin.so.1.1.65         libnvinfer_plugin.so.7         libnvinfer_static.a       libnvparsers.so.7      stubs
libmyelin_executor_static.a         libnvcaffe_parser.a         libnvinfer_plugin.so.7.2.1     libnvonnxparser.so        libnvparsers.so.7.2.1
libmyelin_pattern_library_static.a  libnvcaffe_parser.so        libnvinfer_plugin_static.a     libnvonnxparser.so.7      libnvparsers_static.a
libmyelin_pattern_runtime_static.a  libnvcaffe_parser.so.7      libnvinfer.so                  libnvonnxparser.so.7.2.1  libonnx_proto.a
libmyelin.so                        libnvcaffe_parser.so.7.2.1  libnvinfer.so.7                libnvonnxparser_static.a  libprotobuf.a
libmyelin.so.1                      libnvinfer_plugin.so        libnvinfer.so.7.2.1            libnvparsers.so           libprotobuf-lite.a
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6> ls ./include
NvCaffeParser.h  NvInferPlugin.h       NvInferRuntimeCommon.h  NvInferVersion.h  NvOnnxParser.h  NvUtils.h
NvInfer.h        NvInferPluginUtils.h  NvInferRuntime.h        NvOnnxConfig.h    NvUffParser.h
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6> ls data/
char-rnn  faster-rcnn  googlenet  int8_api  mlp  mnist  movielens  resnet50  ssd
(env) (cmsc) linux9:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6>
```

Build and run the samples

- Samples can be found in http://github.com/NVIDIA/TensorRT/tree/master/samples/opensource.

Installation

Check Installation

sampleMNIST:
https://github.com/NVIDIA/TensorRT/tree/master/samples/opensource/sampleMNIST

- Download MNIST :
  step into ./TensorRT-7.2.1.6/data/mnist and download the dataset
- Compile the sample :
  step into /TensorRT-7.2.1.6/samples/sampleMNIST and then make
- the compiled binary files are in ./TensorRT-7.2.1.6/bin

```
(env) (cmsc) gpu44:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/samples/sampleMNIST> ls
Makefile  README.md  sampleMNIST.cpp
(env) (cmsc) gpu44:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/samples/sampleMNIST> make
../Makefile.config:15: CUDNN_INSTALL_DIR variable is not specified, using /usr/local/cuda-11.0 by default, use CUDNN_INSTALL_DIR=<cudnn_directory> to change.
../Makefile.config:28: TRT_LIB_DIR is not specified, searching ../../lib, ../../lib, ../lib by default, use TRT_LIB_DIR=<trt_lib_directory> to change.
Linking: ../../bin/sample_mnist_debug
Linking: ../../bin/sample_mnist
# Copy every EXTRA_FILE of this sample to bin dir
```

Installation

Check Installation

```
(env) (cmsc) gpu44:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/samples/sampleMNIST> cd ../../bin
(env) (cmsc) gpu44:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/bin> ./sample_mnist
&&&& RUNNING TensorRT.sample_mnist # ./sample_mnist
[11/01/2020-17:55:51] [I] Building and running a GPU inference engine for MNIST
[11/01/2020-17:56:07] [I] [TRT] Some tactics do not have sufficient workspace memory to run. Increasing workspace size may increase performance, please check verbose output.
[11/01/2020-17:56:08] [I] [TRT] Detected 1 inputs and 1 output network tensors.
[11/01/2020-17:56:08] [I] Input:
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@++.  .@@@@@@
@@@@@@@@@@@@#+-      .@@@@@@
@@@@@@@@%:..   :      .@@@@@@
@@@@@@@#.     %@%#.   :@@@@@@
@@@@@@@:    +#@@@=   .@@@@@@@
@@@@@#    .#@@@@@:   =@@@@@@@
@@@@@:   :#@@@@@*   :#@@@@@@@
@@@@@*.:%@@@@+   .%@@@@@@@@@
@@@@@@@@@@@@*    +@@@@@@@@@@
@@@@@@@@@@@@@    +@@@@@@@@@@
@@@@@@@@@@@@=    %@@@@@@@@@@
@@@@@@@@@@@:   +@@@@@@@@@@@@
@@@@@@@@@@+   .@@@@@@@@@@@@@
@@@@@@@@@@-   +@@@@@@@@@@@@@
@@@@@@@@@%   .@@@@@@@@@@@@@@
@@@@@@@@%-  .%@@@@@@@@@@@@@@
@@@@@@@@-   +@@@@@@@@@@@@@@@
@@@@@@@%:  =%@@@@@@@@@@@@@@@
@@@@@@@= .%@@@@@@@@@@@@@@@@@
@@@@@@@# *@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@

[11/01/2020-17:56:08] [I] Output:
0:
1:
2:
3:
4:
5:
6:
7: **********
8:
9:


&&&& PASSED TensorRT.sample_mnist # ./sample_mnist
(env) (cmsc) gpu44:/research/d1/qsun/course/cmsc5743-lab06/TensorRT-7.2.1.6/bin>
```

Basic Usage

Do inference on CUDA GPU

- There's a binary file trtexec in ./TensorRT-7.2.1.6/bin.
- You can use C++ API or Python API.

Steps

ONNX $\rightarrow$ TRT.

- Get model ONNX file.
- Build CUDA engine.
- Prepare inputs.
- Create CUDA context.
- Do inference.
- Process outputs.

Basic Usage

TensorRT

https://docs.nvidia.com/deeplearning/tensorrt/api/python_api/index.html

- tensorrt.Builder
- tensorrt.OnnxParser
- tensorrt.ICudaEngine

PyCUDA

https://documen.tician.de/pycuda/index.html

- pycuda.driver
- pycuda.autoinit