Qualcomm Aimet Efficiency Toolkit Documentation Instructions

KBA-231226181840

1. Setup Envirnment

1.1. Install Nvidia Driver and CUDA

1.2. Install Related Python Library

```
python3 -m pip install –upgrade –ignore-installed pip
python3 -m pip install –ignore-installed gdown
python3 -m pip install –ignore-installed opencv-python
python3 -m pip install –ignore-installed torch==1.9.1+cu111 torchvision==0.10.1+cu111 torchaudio==0.9.1 -f
https://download.pytorch.org/whl/torch_stable.html
python3 -m pip install –ignore-installed jax
python3 -m pip install –ignore-installed ftfy
python3 -m pip install –ignore-installed torchinfo
python3 -m pip install –ignore-installed https://github.com/quic/aimet/releases/download/1.25.0/AimetCommon-
torch_gpu_1.25.0-cp38-cp38-linux_x86_64.whl
python3 -m pip install –ignore-installed https://github.com/quic/aimet/releases/download/1.25.0/AimetTorch-
torch_gpu_1.25.0-cp38-cp38-linux_x86_64.whl
python3 -m pip install –ignore-installed numpy==1.21.6
python3 -m pip install –ignore-installed psutil
```

1.3. Clone aimet-model-zoo

```
git clone https://github.com/quic/aimet-model-zoo.git
cd aimet-model-zoo
git checkout d09d2b0404d10f71a7640a87e9d5e5257b028802
export PYTHONPATH=${PYTHONPATH}:${PWD}
```

1.4. Download Set14

```
wget https://uofi.box.com/shared/static/igsnfieh4lz68l926l8xbklwsnnk8we9.zip
unzip igsnfieh4lz68l926l8xbklwsnnk8we9.zip
```

1.5. Modify line 39 aimet-model-zoo/aimet_zoo_torch/quicksrnet/dataloader/utils.py

```
change
for img_path in glob.glob(os.path.join(test_images_dir, “*”)):
to
for img_path in glob.glob(os.path.join(test_images_dir, “*_HR.*”)):
```

1.6. Run evaluation.

```
# run under YOURPATH/aimet-model-run
# For quicksrnet_small_2x_w8a8
```

```
python3 aimet_zoo_torch/quicksrnet/evaluators/quicksrnet_quanteval.py \
–model-config quicksrnet_small_2x_w8a8 \
–dataset-path ../Set14/image_SRF_4

# For quicksrnet_small_4x_w8a8
python3 aimet_zoo_torch/quicksrnet/evaluators/quicksrnet_quanteval.py \
–model-config quicksrnet_small_4x_w8a8 \
–dataset-path ../Set14/image_SRF_4

# For quicksrnet_medium_2x_w8a8
python3 aimet_zoo_torch/quicksrnet/evaluators/quicksrnet_quanteval.py \
–model-config quicksrnet_medium_2x_w8a8 \
–dataset-path ../Set14/image_SRF_4

# For quicksrnet_medium_4x_w8a8
python3 aimet_zoo_torch/quicksrnet/evaluators/quicksrnet_quanteval.py \
–model-config quicksrnet_medium_4x_w8a8 \
–dataset-path ../Set14/image_SRF_4
```

suppose youwill get the PSNRvaluefor theaimetsimulated model. You can changethe model-config for differentsize ofQuickSRNet, the option is underaimet-modelzoo/aimet_zoo_torch/quicksrnet/model/model_cards/.

2 Add Patch

2.1. Open “Export to ONNX Steps REVISED.docx”

2.2. Skip git commit id

2.3. Section 1 Code

Add whole 1. code under last line (after line 366) aimet-model-zoo/aimet_zoo_torch/quicksrnet/model/models.py

2.4. Section 2 and 3 Code

Add whole 2, 3 code under line 93 aimet-model-zoo/aimet_zoo_torch/quicksrnet/evaluators/quicksrnet_quanteval.py

2.5. Key Parameters in Function load_model

model = load_model(MODEL_PATH_INT8, MODEL_NAME, MODEL_ARGS.get(MODEL_NAME).get(MODEL_CONFIG), use_quant_sim_model=True, encoding_path=ENCODING_PATH, quantsim_config_path=CONFIG_PATH, calibration_data=IMAGES_LR, use_cuda=True, before_quantization=True, convert_to_dcr=True)
MODEL_PATH_INT8 = aimet_zoo_torch/quicksrnet/model/weights/quicksrnet_small_2x_w8a8/pre_opt_weights MODEL_NAME = QuickSRNetSmall MODEL_ARGS.get(MODEL_NAME).get(MODEL_CONFIG) = {‘scaling_factor’: 2} ENCODING_PATH = aimet_zoo_torch/quicksrnet/model/weights/quicksrnet_small_2x_w8a8/adaround_encodings CONFIG_PATH = aimet_zoo_torch/quicksrnet/model/weights/quicksrnet_small_2x_w8a8/aimet_config
Please replace the variables for different size of QuickSRNet

2.6 Model Size Modification
1. “input_shape” in aimet-model-zoo/aimet_zoo_torch/quicksrnet/model/model_cards/*.json 2. Inside function load_model(…) in aimet-model-zoo/aimet_zoo_torch/quicksrnet/model/inference.py 3. Parameter inside function

export_to_onnx(…, input_height, input_width) from “Export to ONNX Steps REVISED.docx”
2.7 Re-Run 1.6 again for exporting ONNX model
3. Convert in SNPE
3.1. Convert
${SNPE_ROOT}/bin/x86_64-linux-clang/snpe-onnx-to-dlc –input_network model.onnx –quantization_overrides ./model.encodings
3.2. (Optional) Extract only quantized DLC
(optional) snpe-dlc-quant –input_dlc model.dlc –float_fallback –override_params
3.3. (IMPORTANT) The ONNX I/O is in order of NCHW; The converted DLC is in order NHWC

Contents

Documents / Resources

	Qualcomm Aimet Efficiency Toolkit Documentation [pdf] Instructions quicksrnet_small_2x_w8a8, quicksrnet_small_4x_w8a8, quicksrnet_medium_2x_w8a8, quicksrnet_medium_4x_w8a8, Aimet Efficiency Toolkit Documentation, Efficiency Toolkit Documentation, Toolkit Documentation, Documentation

References

- download.pytorch.org/whl/torch_stable.html
- github.com/quic/aimet/releases/download/1.25.0/AimetCommon-torch_gpu_1.25.0-cp38-cp38-linux_x86_64.whl
- github.com/quic/aimet/releases/download/1.25.0/AimetTorch-torch_gpu_1.25.0-cp38-cp38-linux_x86_64.whl
- User Manual