QONNX dev meeting

@FastML for Science Workshop
2023-09-28

Agenda

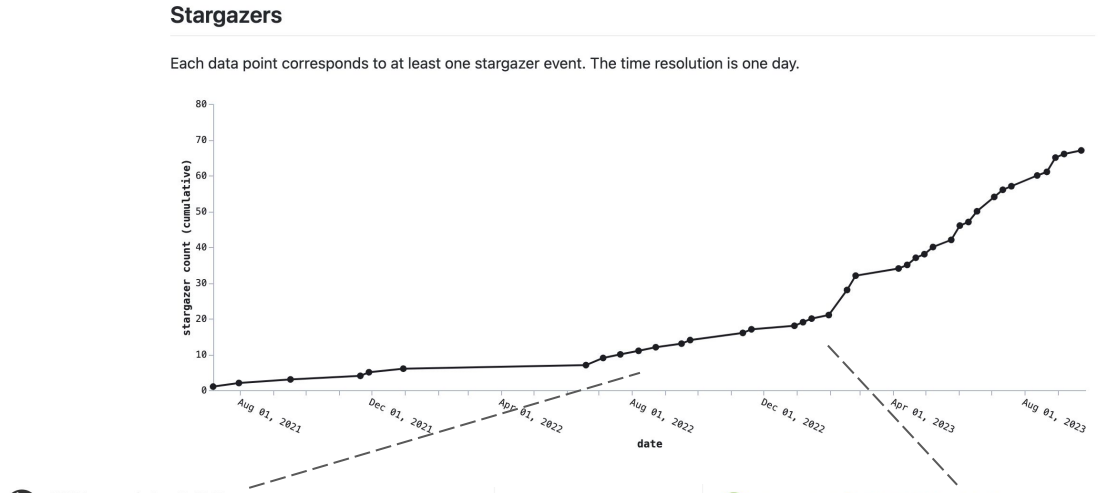
1. Statistics
2. What’s in the repo today?
3. What’s new + what’s coming
4. What would you like to see in qonnx?
    a. Poll results, transformation wiki update
5. <insert more items>

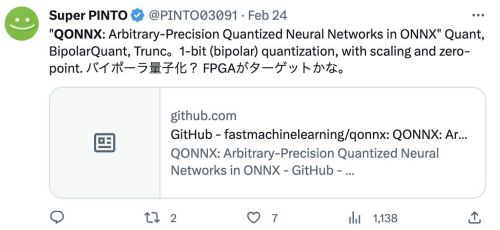
Stats as per 2023-09-28

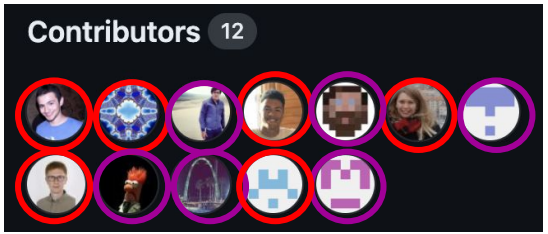
Downloads 71k from pypi + 5.3k form GitHub clones

Fork 24 | Starred 68

3 Open | 55 Closed







FINN hls4ml

8 models in model zoo: JSC, MNIST, CIFAR-10, ImageNet

What's in the repo today?

	python.module.for.import **commandline-util**
executing QONNX models for (slow) functional verifications	qonnx.core.onnx_exec **qonnx-execute**
Cleanup: shape inference, constant folding, cleanup and other transformations (~30 different ones)	qonnx.transformation **qonnx-cleanup**
summarizing the inference cost of a QONNX model in terms of mixed-precision MACs, parameter and activation volume	qonnx.util.inference_cost **qonnx-inference-cost**
Conversion between different ONNX quantization representations (Quant <-> QCDQ <-> QOp in progress)	qonnx.util.convert **qonnx-convert**
Python infrastructure for writing transformations and defining executable, shape-inferencable custom ops	qonnx.transformation.base qonnx.custom_op.base

What’s new recently merged PRs

Channel pruning utilities ✓
#71 by maltanar was merged last month

Fix (utils): Updating matvec accumulator range calculation ✓
#70 by i-colbert was merged last month

Fix: Adding dropped attributes to transposed convolution op ✓
#69 by i-colbert was merged on Aug 15 2 tasks done

[Convert] simplify zeropoint to scalar in QCDQ2Quant if possible ✓
#68 by maltanar was merged on Aug 3

Feat: Adding quantization support for sub-pixel convolution to deconvolution transformation ✓
#67 by i-colbert was merged on Aug 4

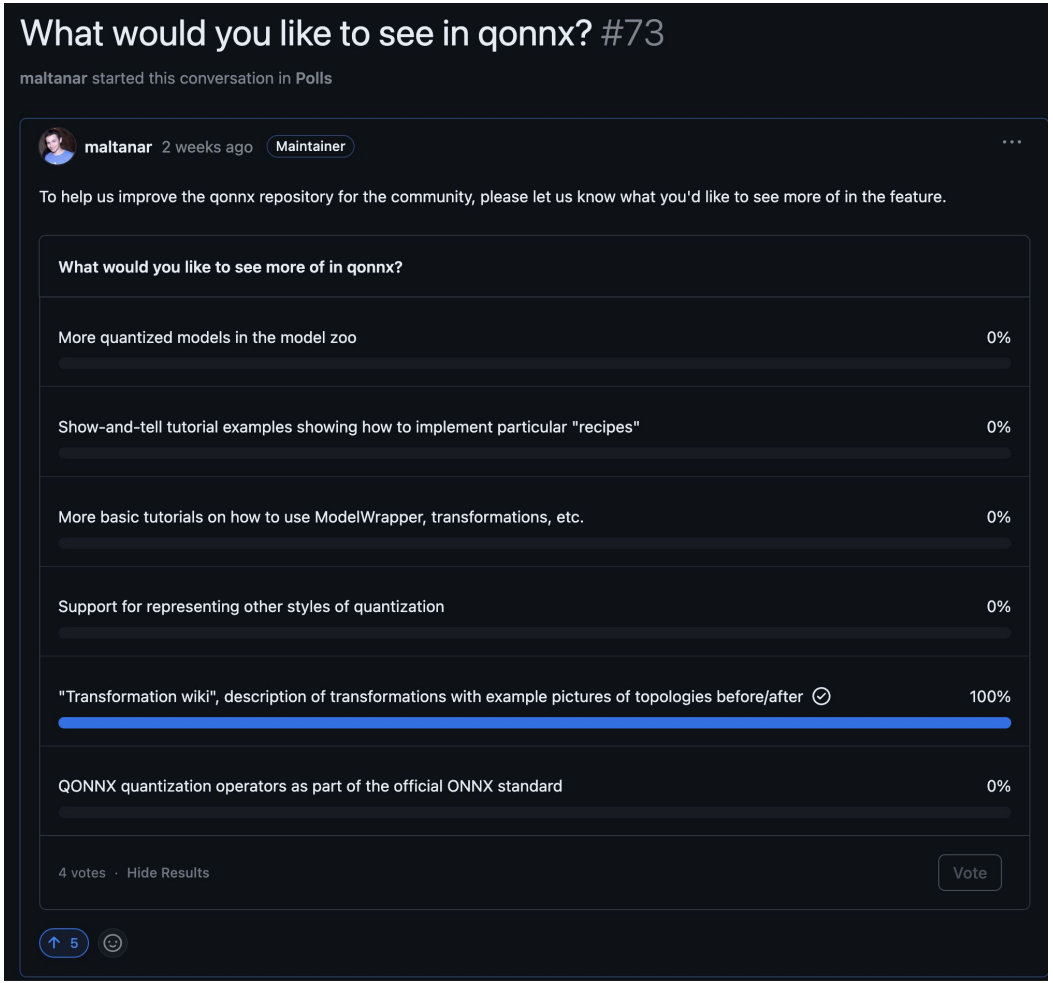
Introduce RemoveUnusedNodes ✓
#66 by maltanar was merged on Jul 19 v0.3.0

Quant/QCDQ converter improvements ✓
#65 by maltanar was merged on Jul 18 v0.3.0

qonnx-exec improvements ✓
#64 by maltanar was merged on Jul 18 v0.3.0

Coming up soon

- QOp conversion (AMD)
- Recurrent nets with Quant (Shashwat/AMD/Trinity College)
- Range analysis (Yaman/AMD)
- Float quantization? (Freddie/Bristol)



Can we do this semi-automatically?

Almost all qonnx transforms have unit tests, they all do something like:

```
new_model = model.transform(BatchNormToAffine())
```

Use `intercepts` package to intercept call to transformations of interest in a special script:

```
intercepts.register(ModelWrapper.transform, handler)
handler:
if transformation in [BatchNormToAffine, trafo2…]:
       Save ONNX model before transformation
       Call the transformation
       Save ONNX model after transformation
```

Execute test suite to generate all before/after ONNX pairs + upload to git

Link to before/after files (+Netron?) from wiki, add descriptions manually

Thank you!