

**INDUSTRY:** special-purpose microprocessors, AI accelerators

#### **PROBLEM STATEMENT:**

The customer specializes in the development and manufacture of special-purpose tensor processing units (TPU) for speedy neural network calculations based on matrix multiplication block, which performs the most resourceintensive calculations at a rate of tens of thousands of operations per cycle. The customer asked us to develop a software complex for a computation graph translator of neural networks from the TensorFlow framework to the mnemonic format of intermediate representation of the TPU model.

#### **SOLUTION:**

We used a basic TVM framework to develop a translator. The translator receives neural networks in the TensorFlow Frozen Graph format processed with quantization service (provided by customer) as input data. Within the project timeframe (18 months, team: software architect and software developers), the following tasks were completed:

- TVM framework back-end for translating neural network into mnemonic format of TPU intermediate representation.
- Optimization of different TVM levels to adjust neural networks to TPU architecture peculiarities.
- Launching neural network calculations on TPU model and TPU hardware implementation using TVM interface.

The translator is run under Ubuntu 18.04.3 LTS, CentOS 7, and Windows 10 operating systems.

The software complex provides translation of testing set neural networks using the following TensorFlow operations:

## Array operations

Concat, ConcatV2, DepthToSpace, MirrorPad, Reshape, Reverse, ReverseSequence, ReverseV2, SpaceToDepth, Depth2Space, Split, SplitV, Squeeze, StridedSlice, Tile, Transpose, Upsample, Flatten, RepeatVector

## **Activation operations**

AvgPool, BiasAdd, BiasAddV1, MaxPool/MaxPoolV2, MaxPool3D, Pad, PadV2, Relu, Relu6, LeakyRelu, Flatten, Reduce\_mean, Sigmoid, Tanh

#### Math operations

Fully Connected, Conv2D, Conv3D, DepthwiseConv2dNative, Conv2D\_transpose, Add, AddV2, BatchMatMul, MatMul, Mul/Multiply, Maximum, Slice, BatchNorm

Within the project scope, a set of testing neural networks to test translator functionality was developed including the following neural networks:

#### **Classification:**

- LeNet5
- AlexNet
- SqueezeNet
- VGG19

RESULT

- Inception\_v1 (GoogLeNet)
- Inception\_v3
  - ResNetV1-50
- ResNetV1-152 ResnetV2-50
- SENet
- EfficientNet
- PNasNet-large
- MobileNet\_v1
- MobileNet\_v2
- PNasNet-small
- Segmentation:
- Yolo-v2 Yolo-v3
- SSDNet
- MobileFaceNets
- VDSR
- Mask-RCNN

# Generative adversarial network (GAN):

- Convolutional GAN
- Deep Convolutional GAN
- StyleGAN
- pix2pix

Besides the readily available neural networks, we also developed special-purpose testing networks for all supported input operation and basic combinations with them.

The customer got a special-purpose neural networks translation software for MCU models that had been prepared for release. Therefore, the customer got a translator featuring neural network compiling and launching in the TensorFlow format on the customer's models and hardware.

# TOOLS & TECHNOLOGIES:

TVM, TensorFlow, C/C++, Python



