AI Compiler @ Alibaba

Presenting the work of many people!

Xiaoyong Liu

PAI (Platform of AI)
Alibaba Cloud Intelligence
How TVM is used @ Alibaba

• An End-to-End Deep Learning Compiler
  ➢ Empower AI service
  ➢ Generate high performance operators
    • subgraph & kernel
    • heterogenous computing

• An Optimizer & Compiler
  ➢ Enable chips such as CPU, GPU, DSP & etc., potentially FPGA, AI Chips.
  ➢ Deploy algorithm automatically

• All Scenarios
  ➢ Cloud, Edge & IoT
  ➢ Training & Inference
TVM + AIService : PAI-Blade
Things We Experienced

- Current approach is too much engineering effort, difficult for platform service

- TVM is good at
  - To generate high-performance computing intensive kernels
    - Automatic is the key
  - Heterogenous hardware friendly, if ISA is provided
    - Performance portability
  - Software Architect friendly to Auto TVM / Schedule…
  - Whole-Graph Optimization

- Challenges
  - Easy of deployment, including coverage, quality & compatibility
    - Correctness, Performance & easy of new device enabling
  - Systems don’t interop
  - Maturity/Standardization …
Contributed to TVM Community

- **Automatic Tensor Core Scheduling**
  - Nvidia tensor core in V100/T4

- **Schedule Algorithm enablement, as Batch Matmul and etc.**
  - Know what/how

- **Support TFLite models**
  - Automatically

- **C++ RPC Server**
  - Tuning your program in embedded environment without python
Ongoing Effort to Community

- **Automatic Tensor Core Scheduling Enhancement**
  - vthread supporting

- **Operators: New Ops in Relay / TVM**
  - HashTable, Embedding…

- **Standardize GraphRuntime Exports Into a Single DLL**
  - A way to unify runtime models exports
Product-driven TVM enhancement

- Brings online inference service
- Compiles heterogenous hardware at cloud & edge
  - Nvidia server GPU
    - V100/T4 on FP16/INT8/INT4/INT1
  - Intel X86 Server CPU
    - on INT8/FP32/BF16
  - ARM64 CPU
    - on INT8 / FP32
  - ARM32 CPU
- Enhances Infrastructure
  - HIFI4 DSP
  - Hexagon DSP
  - PowerVR GPU
  - Intel GPU

Any general solution is planned to contribute back to TVM Community!
TVM produced more performance

• VS
  - Chip supplier’s latest manually optimized high performance library
  - Assembly-level optimized edge machine learning framework

• Optimized to gain decent performance on various products
  - Server Nvidia GPU
    - Automatic TC Scheduling + tensorization + tensorcore
  - Edge Arm64
  - IoT Arm32
## Performance on V100 (FP16)

<table>
<thead>
<tr>
<th>M, N, K</th>
<th>cuBLAS TensorCore</th>
<th>TVM TensorCore</th>
<th>speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>512, 16, 512</td>
<td>7.7470us</td>
<td>5.2570us</td>
<td>1.47X</td>
</tr>
<tr>
<td>512, 32, 512</td>
<td>8.0140us</td>
<td>6.0220us</td>
<td>1.33X</td>
</tr>
<tr>
<td>512, 64, 512</td>
<td>8.7530us</td>
<td>6.2390us</td>
<td>1.40X</td>
</tr>
<tr>
<td>512, 128, 512</td>
<td>9.0290us</td>
<td>7.1610us</td>
<td>1.26X</td>
</tr>
<tr>
<td>256, 256, 256</td>
<td>6.9380us</td>
<td>4.5930us</td>
<td>1.51X</td>
</tr>
<tr>
<td>1024, 32, 512</td>
<td>8.3320us</td>
<td>6.3770us</td>
<td>1.30X</td>
</tr>
<tr>
<td>2048, 32, 512</td>
<td>9.0640us</td>
<td>7.5070us</td>
<td>1.21X</td>
</tr>
</tbody>
</table>
Performance on T4

- Cublas INT8
- TVM INT8
- TVM INT4
- TVM INT1

Speedup

- (512, 64, 512)
- (512, 32, 512)
- (512, 16, 512)

cublas baseline
AliOS enhances TVM on vehicles

- To accelerate NLU and AR-Nav models
- ARM64 CPU performance on INT8 / FP32
  - NHWC/img2col+pack/no tensorize&co-optimized with llvm
  - Planning to contribute back to community
- Hexagon DSP
  - vrmpy tensorize/llvm-codegen
  - Could run end-to-end Mobilenet V2 INT8 model
- Intel GPU
  - Schedule algorithm
  - Boost 1.6X performance of Lanenet model
Performance on ARM64 INT8

Performance Comparison @ rasp 3b+ AARCH64

- MobileNetV1
- MobileNetV2
- LaneNet

- TFLite 1 core
- TFLite 4 core
- QNNPACK 1 core
- QNNPACK 4 core
- TVM 1 core
- TVM 4 core

Alibaba Group
Performance on ARM64 FP32

Performance Comparison AARCH64

- **Mobilenet V1**
  - TVM / MNN @ A53: 1.07
  - TVM / MNN @ A72: 1.17

- **Mobilenet V2**
  - TVM / MNN @ A53: 1.03
  - TVM / MNN @ A72: 1.13
AI Labs Compiles TMallGenie Models

- **ARM32 CPU**
  - Overflow-Aware Quantization (INT16 = INT8 * INT8)
  - GEMM Tensorize

- **HIFI4 DSP**
  - GEMM Tensorize, 10X speed up

- **PowerVR GPU**
  - Schedule Algorithm
Performance

CPU: MTK8167S (ARM32 A35 1.5GHz)
Model: MobileNetV2_1.0_224
A DL Compiler in T-HEAD SoC

- TVM has been integrated into WuJian(无剑) SoC toolchain
- Support Caffe Frontend
  - Tested pass alexnet / resnet 50 / mobilenet v1 / mobilenet v2 / …

![Diagram showing integration of TensorFlow, Caffe, TVM, T-HEAD NN, LLVM, and WuJian SoC]
TVM Roadmap @ Alibaba

- Keep contributing general effort back to community

- **Auto Schedule** ("with Berkeley team")
  - Auto* is the key to build machine-learning-powered system

- Interpolate with top frameworks

- **Auto heterogenous hardware placement in system level**

- **Infra Maturity**
  - Completeness & Seamless Deployment, as quantization, model compatibility

- **Workload Characterization**
  - To improve the key workloads within community

- **AI Service & Operators**
  - More chips, more models
Takeaways

• A golden age of deep learning compiler

• Industry-grade deep-learning compilation solution is still in evolution

• We are working to contribute to TVM
  ➢ Development & Research

• Welcome to join us to contribute to TVM together
  ➢ Xiaoyong Liu (xiaoyong.liu@Alibaba-inc.com)
Thank you!