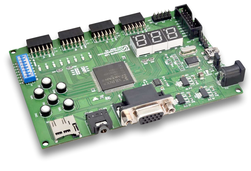
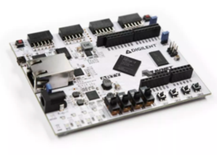
µTVM: Deep Learning on Bare-Metal Devices

Logan Weber, Pratyush Patel, and Tianqi Chen

The Move Towards the Edge

With the astounding success of machine learning in general, many researchers and practitioners have turned their attention to the edge.





FPGA

Arduino

Bare-metal devices are commonly found on the edge, because they are cheap and energy-efficient.

The Problem

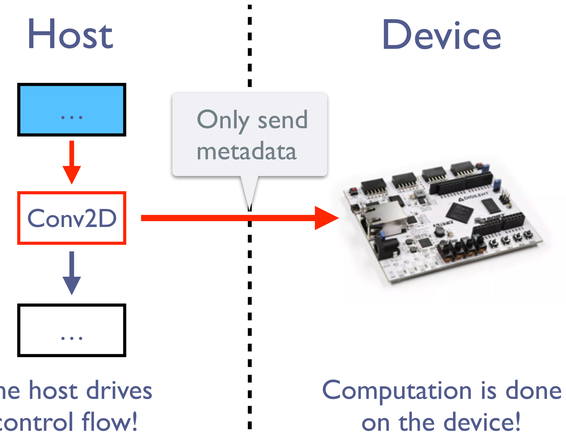
Programming these devices is extremely difficult due to:

- Resource constraints
- Lack of compute power
- Lack of on-device memory management
- Restricted language and runtime support
- Tedious debugging

µTVM's Approach

- Plug directly into TVM as a backend
- Use the compiler to emit code for the device
- Gain access to TVM models and optimization

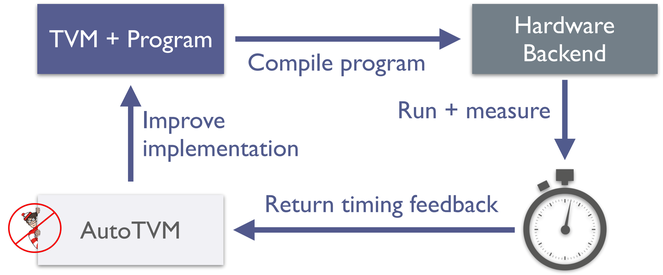
Execution Model



The host drives control flow!

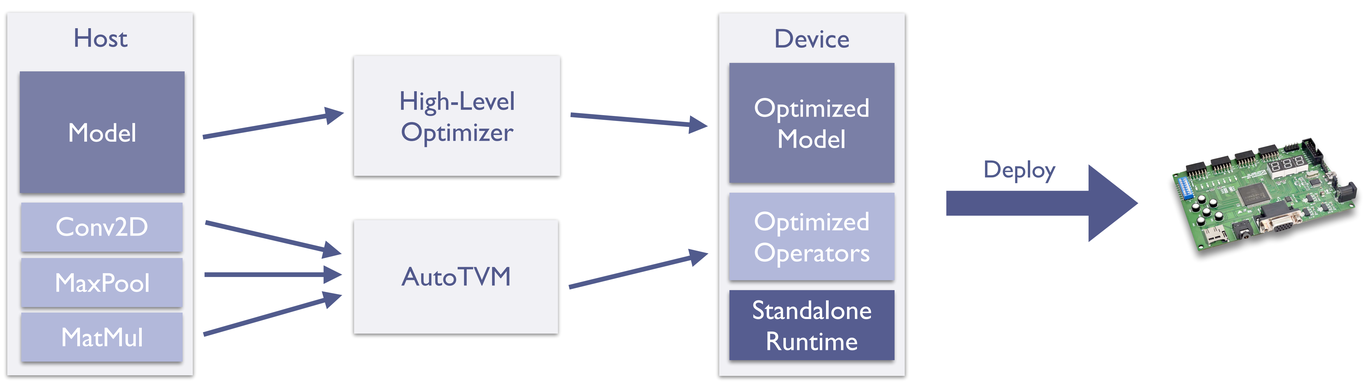
Computation is done on the device!

AutoTVM Compatibility



The current execution model is slow, but it allows us to use TVM's automatic tensor program optimizer.

The End Goal



Contact and Acknowledgements

Logan Weber Pratyush Patel Tianqi Chen

{weberlo, patelp1, tqchen}@cs.uw.edu

PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

uwplse.org • sampl.cs.washington.edu • tvm.ai

This work is supported by the Semiconductor Research Corporation (SRC) and DARPA

adacenter.org @ADA_Center

SRC JUMP ada Applications Driving Architectures