Full-stack evaluation of Machine Learning inference workloads for RISC-V systems

Debjyoti Bhattacharjee, Anmol Anmol, Tommaso Marinelli, Karan Pathak, Peter Kourzanov

imec, Kapeldreef 75, 3001 Leuven, Belgium. {first}.{last}@imec.be

Know more about us:

INTRODUCTION

With the increased demand for **machine-learning** (ML) and **deep-learning** (DL) powered applications, it is important to evaluate their impact on **existing and future architectures**. Architectural simulators like **gem5** provide a convenient way to run workloads on a modeled system, providing performance data that are close to reality. The goals of this work are multiple:

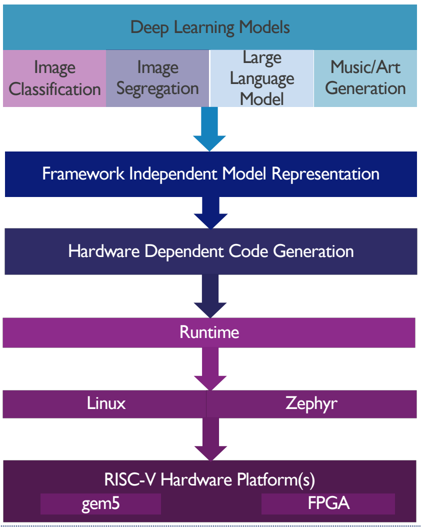
- We aim to assess the **performance** of machine learning **inference workloads** on RISC-V architectures using the gem5 simulator.
- The ML/DL workloads are lowered through **IREE** (Intermediate Representation Execution Environment), an open-source framework to compile the models and distribute their execution across the available hardware resources.
- We leverage an open-source compilation toolchain based on Multi-Level Intermediate Representation (**MLIR**).

BENCHMARKING FLOW

- The IREE flow is used to convert ML/DL models from typical formats (like ONNX or TensorFlow) to **framework-independent** intermediate representations (IR).
- IREE also generates **execution/synchronization logic** to distribute and control the tasks on specific hardware platforms.
- The applications are packed inside an image and singularly executed under a **minimal operating system**, immediately after boot.
- The platform is simulated using a composite environment, consisting of **gem5** for the core and caches, and **SST** (Structural Simulation Toolkit) for the main memory.

Some specific details about this work:

- The reference CPU is a **RISC-V RV64GC** single core processor, in two flavours (in-order or out-of-order).
- The OS is **Linux** and the C standard library is **GLIBC.**
- A **heterogeneous set** of models has been selected.



Deep Learning Models

Image Classification

Image Segregation

Large Language Model

Music/Art Generation

Framework Independent Model Representation

Hardware Dependent Code Generation

Runtime

Linux

Zephyr

RISC-V Hardware Platform(s)

gem5

FPGA

EXPERIMENTAL SETUP

Attribute	Type/version
ISA	rv64gc
Core Type	MinorCPU, O3CPU
Core Freq.	2 GHz
L1 Cache	64KB, 4-way
L2 Cache	8MB, 4-way
DRAM Type	simpleMem
DRAM Size	3GB
DRAM Freq.	1 GHz
Kernel	Linux v6.6.20
Bootloader	OpenSBI v1.4
IREE Version	20230209.43
gem5 version	v23.1 develop

Tab 1: Architecture and Toolchain configuration

Task	Benchmark	Size (MB)
Segmentation	Deeplab V3	2.70
Segmentation	Densenet	41.15
TextDetection	East	23.03
Vision	Efficientnet lite0	4.39
LargeLanguageModel	GPTTwo	472.82
Stylization	Imagestylization	9.00
Classification	Inception V4	162.77
CreativeAI	Imagenet	2.88
DepthEstimation	Midas	63.26
DigitRecognition	MNIST(Lenet5)	1.15
Classification	Mobilenet V1	4.24
Classification	Mobilenet V2	13.30
PoseEstimation	Posenet	12.88
Classification	Resnet (50)	23.52
Classification	Resnet 50	98.53
Classification	Squeezenet	1.66
ObjectDetection	SSD Mobilenet V1	6.65

Tab 2: Benchmarks used for evaluation

EXPERIMENTAL RESULTS

- Fused multiply accumulate operations and memory operations dominate neural network inference

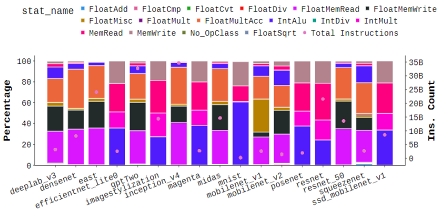


Fig 1: Instruction mix for each ML workload

- Large models benefit the most from using an OoO CPU compared to in-order baselines

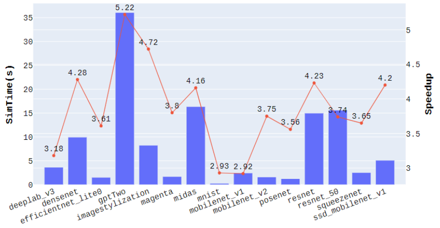


Fig 2: Performance of the workloads running on in-order CPU (Minor CPU) and speedup on running upon O3 CPU

- The reuse of weights in neural networks helps in reducing cache misses at L2

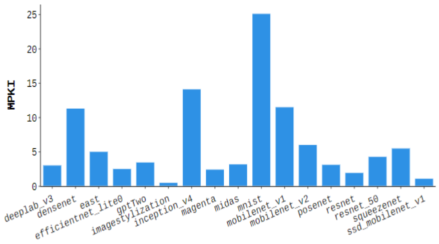


Fig 3: Miss Per Kilo Instruction (MPKI) observed at L2 cache.

- Gem5, being a single threaded simulator takes a long time to simulate the workloads

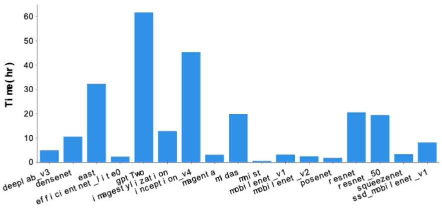


Fig 4: Time (in hours) for simulation of workload on a x86 host.

CONCLUSION AND FUTURE WORK

The key points of the work are:

- We have set up a **workflow** to **evaluate the performance** (and other execution metrics) of ML/DL workloads lowered through IREE.
- The benchmarks have been tested on a **simulated RISC-V platform** under Linux, using specialized tools (gem5 and SST).

The results show that:

- The **out-of-order** model is **significantly faster** than the in-order (5.22x)
- Most instructions are **memory accesses**
- Benchmarks with a **high L2 MPKI** are the ones with **less instructions** (less data reuse)

Some potential directions for future work include:

- Analysis of **additional benchmark suites** (e.g. MLPerf)
- Transition from Linux to **more lightweight operating systems** (e.g. Zephyr), which are better suited for embedded platforms
- **Validation** of simulation data against commercial RISC-V implementations or FPGA softcores
- Evaluation of the impact of the RISC-V **vector extension** (RVV 1.0)
- Simulation of more complex scenarios (e.g. **multi-core** with multiple or parallel workloads)

REFERENCES

[1] F. Bellard, “QEMU, a fast and portable dynamic translator.,” in USENIX annual technical conference, FREENIX Track, vol. 41, pp. 10–5555, California, USA, 2005.

[2] “Spike RISC-V ISA Simulator.” https://github.com/riscv-software-src/riscv-isa-sim. Accessed: 2024-03-12.

[3] J. L. Power and et al., “The gem5 simulator: Version 20.0+,” CoRR, vol. abs/2007.03152, 2020.

[4] C. Lattner and et al., “MLIR: Scaling compiler infrastructure for domain specific computation,” in 2021 (CGO), pp. 2–14, IEEE, 2021.

[5] C. Lattner and V. Adve, “LLVM: A compilation framework for lifelong program analysis & transformation,” in CGO 2004., pp. 75–86, IEEE, 2004.

[6] H.-I. C. Liu, M. Brehler, M. Ravishankar, N. Vasilache, B. Vanik, and S. Laurenzo, “TinyIREE: An ML execution environment for embedded systems from compilation to deployment,” IEEE Micro, vol. 42, no. 5, pp. 9–16, 2022.