Background

Accelerate ML Performance and Simplify Development for PyTorch with OpenXLA

ML development is often stymied by incompatibilities between frameworks and hardware, forcing developers to compromise on technologies when building ML solutions.

OpenXLA is an open-source ML compiler ecosystem co-developed by Alibaba, Amazon Web Services, AMD, Apple, Arm, Google, Intel, Meta, NVIDIA, and more.

OpenXLA will accelerate and simplify ML development by defragmenting the ML stack across front-end frameworks and hardware backends.

Goals

- Build an industry-standard ML compiler based on a shared vision
- Grow a robust OSS ecosystem that encourages community engagement and collaboration
- Encourage the adoption of OpenXLA across the ecosystem
- Make it easy for developers to achieve optimal performance, hardware optionality, and great usability for their workloads

Community Values

Equal footing: Individuals contribute on an equal footing regardless of their affiliation, and the project provides clear pathways to participation. Those who invest time and energy are leading the project.

Consensus-driven decisions: Decisions are made inclusively, with consistent rules and processes for coming to those decisions.

Transparency: Decision outcomes are shared publicly, with clear rationale. As much as possible, decisions happen in the open with ample opportunity for community feedback.

Code of conduct: There are consistent expectations for how community members should behave, regardless of their position in the community.

History

In 2017, Google launched XLA. XLA uses JIT compilation techniques to analyze a model graph, fuse multiple ops together and emit efficient native machine code for a wide range of hardware devices. Unique XLA features like GSPMD scale workloads and automate parallelization of ML computations. XLA is leveraged by companies like Meta, AWS, Hugging Face, Google, and more.

MLIR

In 2019, Google open-sourced MLIR, a generalizable compiler infrastructure that enables the progressive lowering of abstractions and the development of compilers like Flang and CIRCT. OpenXLA will mesh into and extend the MLIR ecosystem.

StableHLO

- A portable ML compute operation set that makes frameworks easier to deploy across different hardware options.
- At the moment, XLA doesn’t have a guaranteed stable API: neither HLO nor MHLO support versioning or promise compatibility guarantees.
- StableHLO stabilizes HLO & MHLO and enhances them with additional functionality, including dynamism, quantization, and sparsity.

Evolving the XLA compiler

OpenXLA enables developers to...

- Compile models across leading ML frameworks such as PyTorch, TensorFlow, and JAX
- Leverage powerful optimizations like weight update sharding and kernel fusion
- Scale training workloads through SPMD and spatial partitioning
- Obtain support for quantization & mixed precision, sparsity, and more
- Generate code for performant execution across hardware devices including GPUs, CPUs, and ML accelerators like Google TPUs and AWS Trainium

... with much more on the horizon.

Get Involved

- Join community discussions on GitHub and Discord
- Visit our YouTube channel for meeting recordings
- Read more about OpenXLA from NVIDIA GTC, Intel Innovation, and Google Cloud Next coverage.

References


Acknowledgements

Magnus Hyttsten, Stella Laurenzo, Andrew Leaver, Robert Hunt, Amber Huffman, Scott Kulchycki, Shantu Roy, Allen Hutchison