AI use cases with oneDNN on Arm

Ashok Bhat

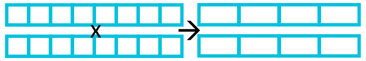
March 2025

AI use-cases with oneDNN on Arm

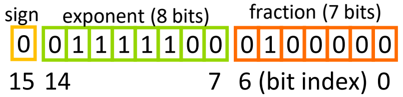
Agenda

- ML hardware features in Arm Neoverse
- Kleidi technology to enable usage of HW features
- Integration of Kleidi technology via oneDNN
- Performance impact
- oneDNN – What is working well?
- oneDNN – What can be better?

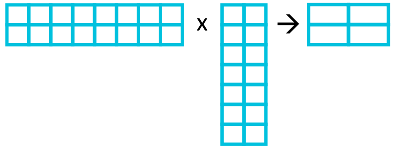
Arm Neoverse has features to accelerate AI



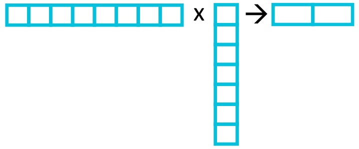
Int8 to Int16 Wide Multiply



BFloat16 Format



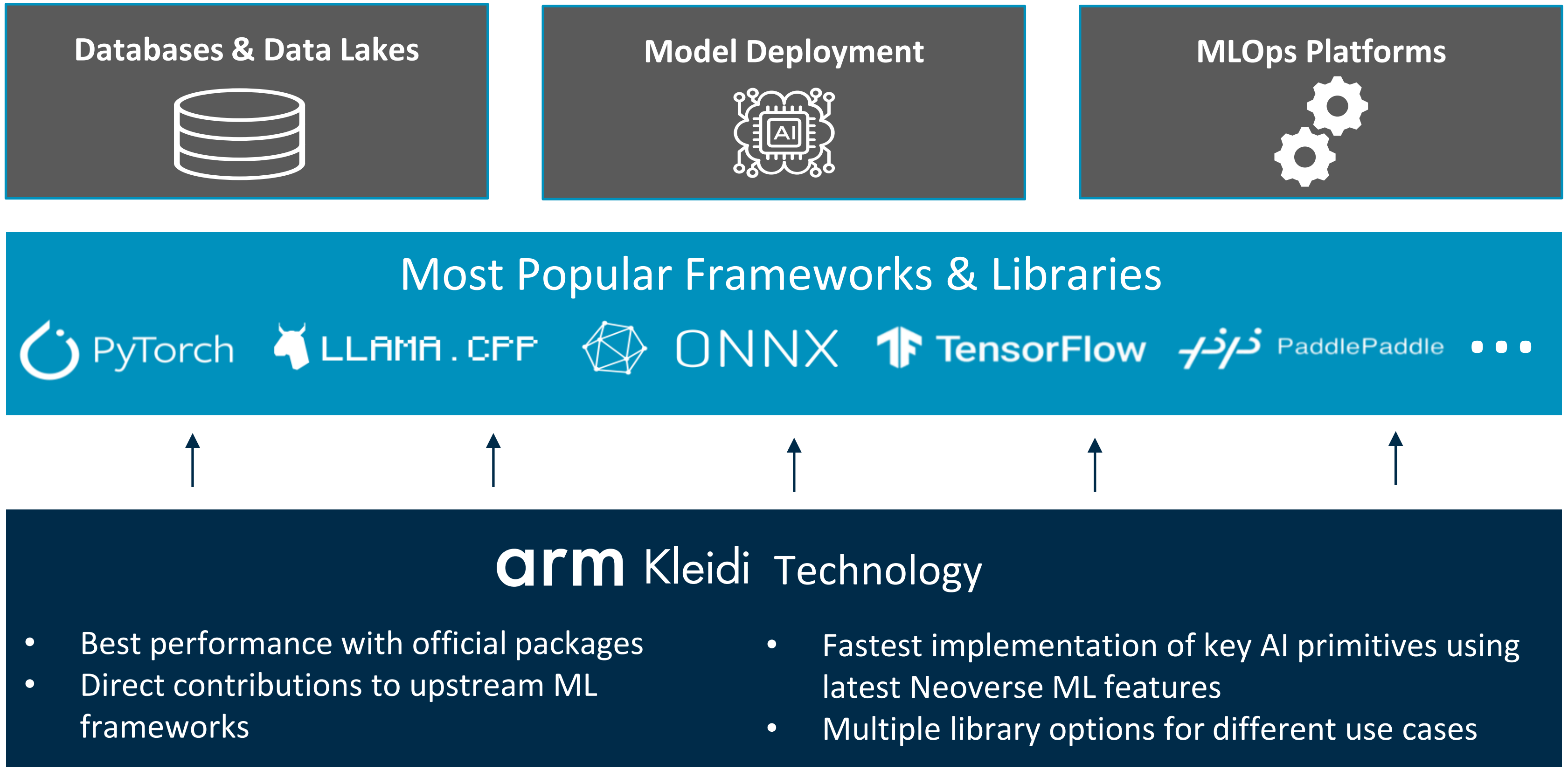
Matrix Multiply



Int8 Dot Product

Software Stack accelerated by Kleidi Technology

arm Kleidi



Databases & Data Lakes

Model Deployment

MLOps Platforms

Most Popular Frameworks & Libraries

PyTorch LLAMA.CPP ONNX TensorFlow PaddlePaddle ...

arm Kleidi Technology

- Best performance with official packages
- Direct contributions to upstream ML frameworks
- Fastest implementation of key AI primitives using latest Neoverse ML features
- Multiple library options for different use cases

ML Libraries with Arm Kleidi technology

Multiple library options for different use-cases

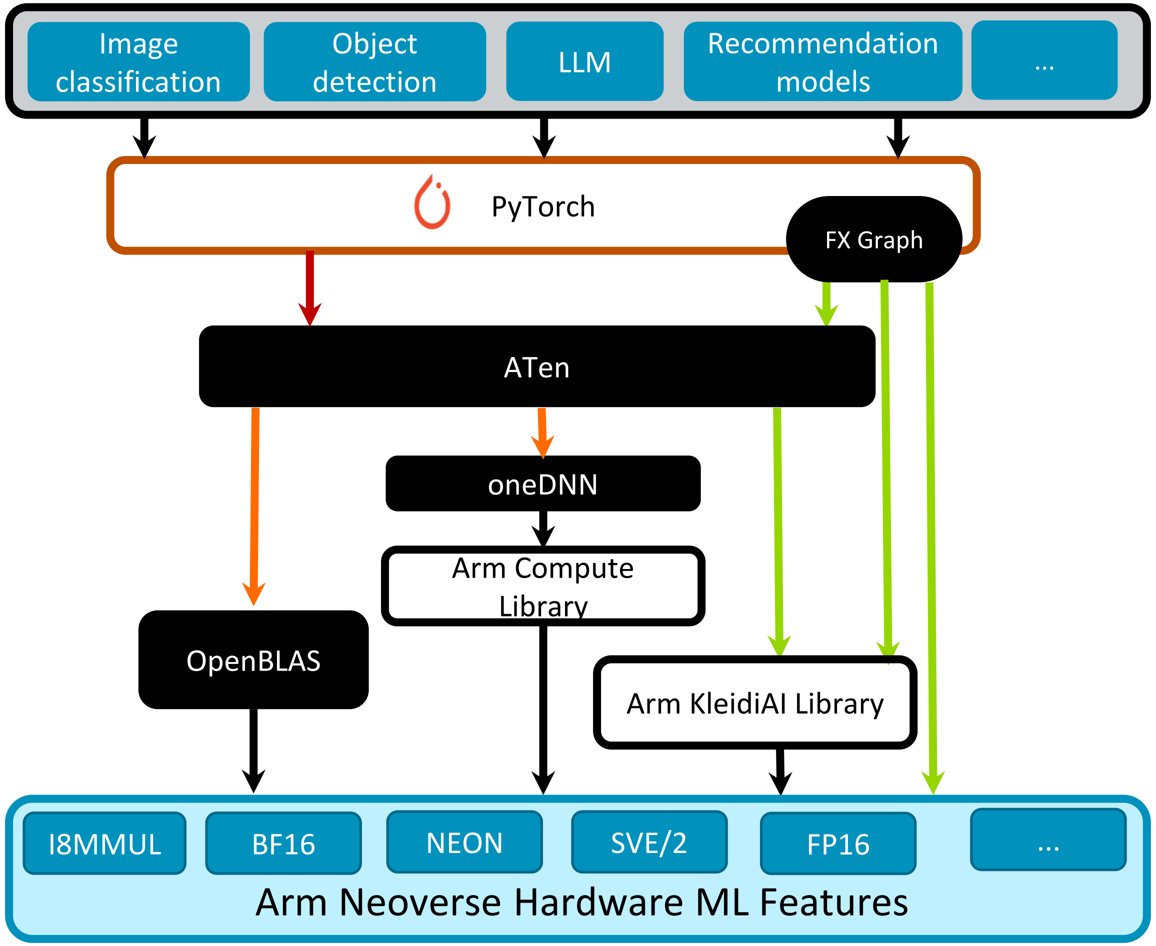
Arm KleidiAI Library

- Optimized low-level micro kernels for Arm CPUs
- Designed for generative AI use-cases
- For frameworks with existing infrastructure for runtime and memory management

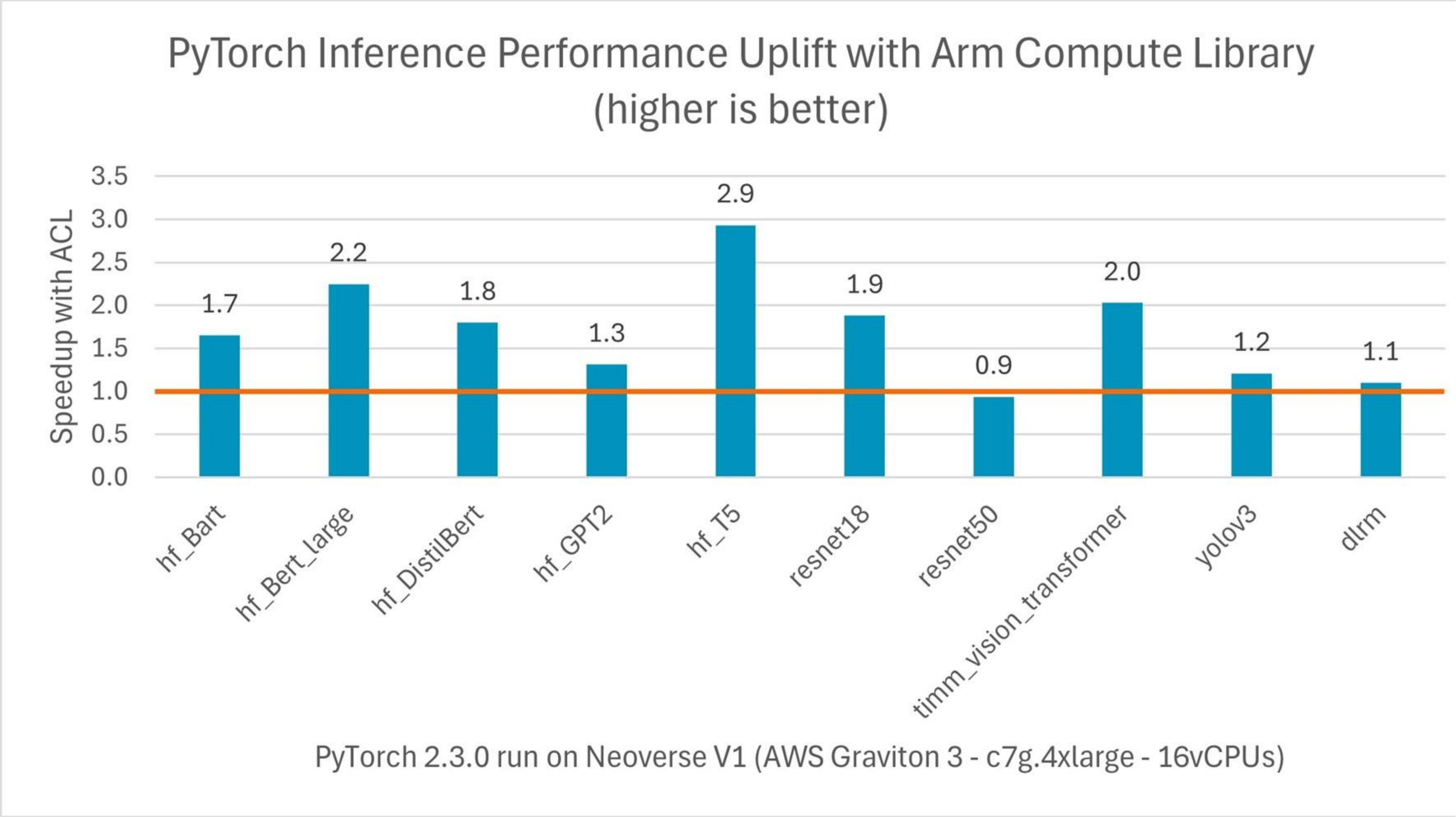
Arm Compute Library

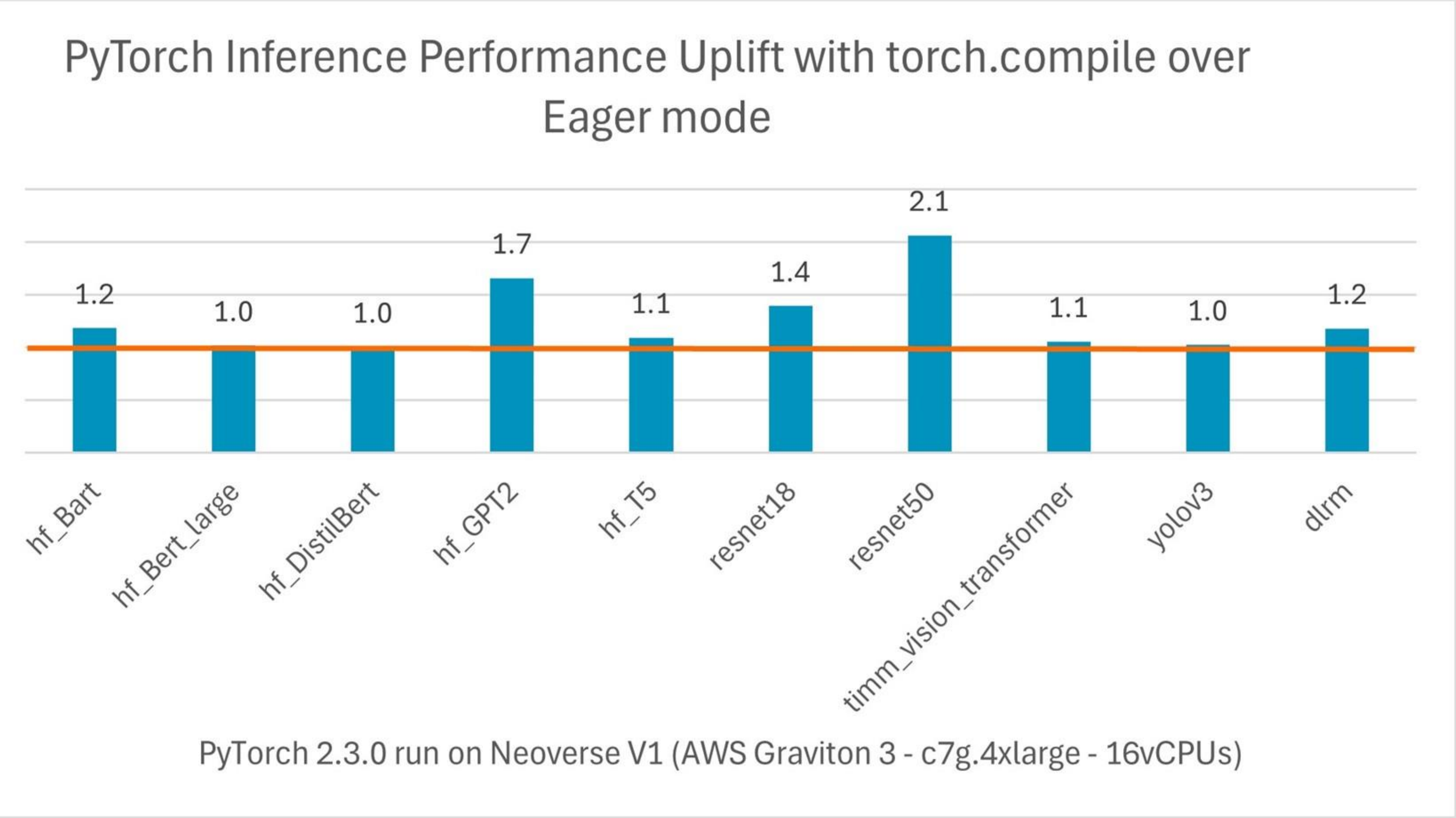
- Optimized high-level ML operators for Arm CPUs
- Designed for traditional ML use-cases like vision, NLP
- For frameworks that delegate model inference computation entirely

Faster PyTorch Inference using Arm Kleidi



- PyTorch uses open-source Arm Compute Library and KleidiAI Library for faster inference
- For traditional use-cases (NLP, Vision, Recommender), operators in ACL is used (via oneDNN)
- For GenAI inference, INT4 quantized kernels in KleidiAI library is used
- Both libraries rely on AWS Neoverse ML features like BF16


PyTorch Inference Performance Uplift with Arm Compute Library
(higher is better)
Speedup with ACL
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
1.7
2.2
1.8
1.3
2.9
1.9
0.9
2.0
1.2
1.1
hf_Bart
hf_Bert_large
hf_DistilBert
hf_GPT2
hf_T5
resnet18
resnet50
timm_vision_transformer
yolov3
dlrm
PyTorch 2.3.0 run on Neoverse V1 (AWS Graviton 3 - c7g.4xlarge - 16vCPUs)



PyTorch Inference Performance Uplift with torch.compile over Eager mode

PyTorch 2.3.0 run on Neoverse V1 (AWS Graviton 3 - c7g.4xlarge - 16vCPUs)

Arm’s experience with oneDNN

What’s working well?

- Significant Contributions:
  - Arm is the major contributor to oneDNN after Intel.
- Effective Collaboration:
  - There is good collaboration between the oneDNN and Arm teams, with excellent response times from the oneDNN team.
- Continuous Integration
  - The oneDNN Arm CI is functioning well.
- oneDNN used in multiple projects
  - TensorFlow, PyTorch, …

Arm’s experience with oneDNN

What can be improved?

- Enhance coordination with teams integrating oneDNN into other projects
  - such as PyTorch, JAX/OpenXLA, etc.
- Timely updates
  - Reduce delays in updating oneDNN in key projects.
- Involvement in major decisions:
  - Ensure Arm is kept in the loop for major decisions, including:
    - Design changes
    - Integration strategy changes
    - Release dates
- Increase community participation

Merci

Danke

Gracias

Grazie

谢谢

ありがとう

Asante

Thank You

감사합니다

धन्यवाद

Kiitos

شكرًا

ধন্যবাদ

תודה

ధన్యవాదములు

Köszönöm