Accelerate AI with Intel® Advanced Matrix Extensions

For the latest version of this guide, see Intel Advanced Matrix Extensions Overview.
Post your questions to Intel DevHub discord or AI Tools forum.

Intel® Advanced Matrix Extensions (Intel® AMX) accelerates deep learning fine-tuning and inference on Intel® Xeon® Scalable processors. Intel AMX is built into every core on 4th and 5th Gen Xeon processors (formerly codenamed Sapphire Rapids & Emerald Rapids), accelerating bfloat16 (BF16) and INT8 data types.

Get started with Intel AMX

Intel AMX can deliver up to 10x generational performance gains[1] for AI workloads. It is enabled in Intel 4th Gen Xeon Scalable processors available through OEMs, partners, or hosted on cloud service providers such as:

Cloud Service Provider	G	aws	IBM Cloud	Alibaba	Bai百度	More to be announced
Intel AMX launch	GCP-C3	C7i, M7i, R7i	8474c	G8i	{GCM}6	

To learn more, see the Tuning Guide for AI on 4th Gen Intel Scalable Processors.

Preparing the model for Intel AMX

For AMX to accelerate your deep learning model, it needs to be in BF16 or INT8 format. You can convert your model to this optimized form using auto-mixed precision for BF16 or quantization for INT8, either natively in your framework (e.g. PyTorch* or TensorFlow*) or with open-source tools from Intel which have additional features.

BF16 is an easy conversion and will generally preserve accuracy. INT8 is a more efficient data type, and you can use Intel's open-source compression tools to preserve accuracy.

BF16 on PyTorch

```
with torch.autocast(device_type="cpu", dtype=torch.bfloat16):
    for input in data:
        # Runs the forward pass with autocasting
        output = model(input)
```

Example recipe

Mixed-precision documentation

BF16 on TensorFlow

```
export TF_SET_ONEDNN_FPMATH_MODE=BF16
```

Get Started Guide & mixed-precision documentation
Convert by setting an environment variable (for v2.13+)

Automatic BF16 with OpenVINO Runtime

OpenVINO™ Runtime, a component of the OpenVINO™ toolkit, is an open source AI deployment library. It will automatically convert eligible models to BF16 when Intel AMX is present (v2023+). OpenVINO can take in TensorFlow, PyTorch, and ONNX models and optimize for accelerated, centralized deployment. Read the Get Started Guide and see examples here.

INT8 Quantization

You can convert your model to the optimized INT8 format within its native framework (PyTorch, TensorFlow, ONNX Runtime*, etc.). Intel also provides open-source tools (Hugging Face* Optimum, OpenVINO NNCF, and Intel Neural Compressor) for quantization with additional features to preserve accuracy.

Notices & Disclaimers: Performance varies by use, configuration, and other factors. Learn more at www.Intel.com/PerformanceIndex. Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure. Your costs and results may vary. Intel technologies may require enabled hardware, software, or service activation.

Performance Claims

[1] https://www.intel.com/content/www/us/en/developer/topic-technology/artificial-intelligence/performance.html