Intel® Network Builders

The secrets to applying AI in Cybersecurity

David Lu (david.lu@intel.com)

intel®

Agenda

- Cybersecurity challenges in the cloud
- AI is the secret to solving cybersecurity
- What is Intel Network AI Offering?
- Intel Network Security AI Success Stories

Cybersecurity Challenges in the Cloud

Stopping Attacks Harder than Ever

Rising Evasive Command and Control
Attackers leverage highly customizable tooling to evade traditional technologies

Signatures ONLY prevent known threats
The attack must be observed first in the wild

Ransomware

Anomaly Attacks

Malware

Phishing

Unknown Attacks

C2 derived from hack tools

Attackers encrypting their network communications
Malware campaigns use encryption to conceal C2 activity

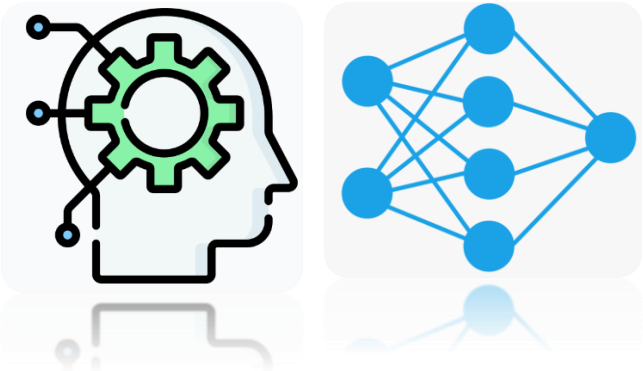
Employing tools to evade signature defenses
struggle to keep pace with the changing threat

AI is the Secret to Solve Cybersecurity

Stops Evasive Threats Inline

Best-In-Class prevention of known threats

detects today's most challenging unknown attacks



Continuous Model Validation/Training

always up to date and easily adopt new capabilities over time

Deep Learning on Live Traffic

Brings next-level prevention inline to def end the initial target

Why is Deep Learning not Widely Adopted?

Platform does matter

model quantization/precision does matter

Deep learning framework does matter

model does matter

- Legacy HW + SW conclusion: They think the CPU is too slow to meet the requirements.
- Developers use old CPUs without Deep Learning boost technologies
- Not good HW and SW out-of-box experiences
- Developers lack knowledge of how to improve deep learning performance

Solutions to Fill the Gap to Adopt Deep Learning

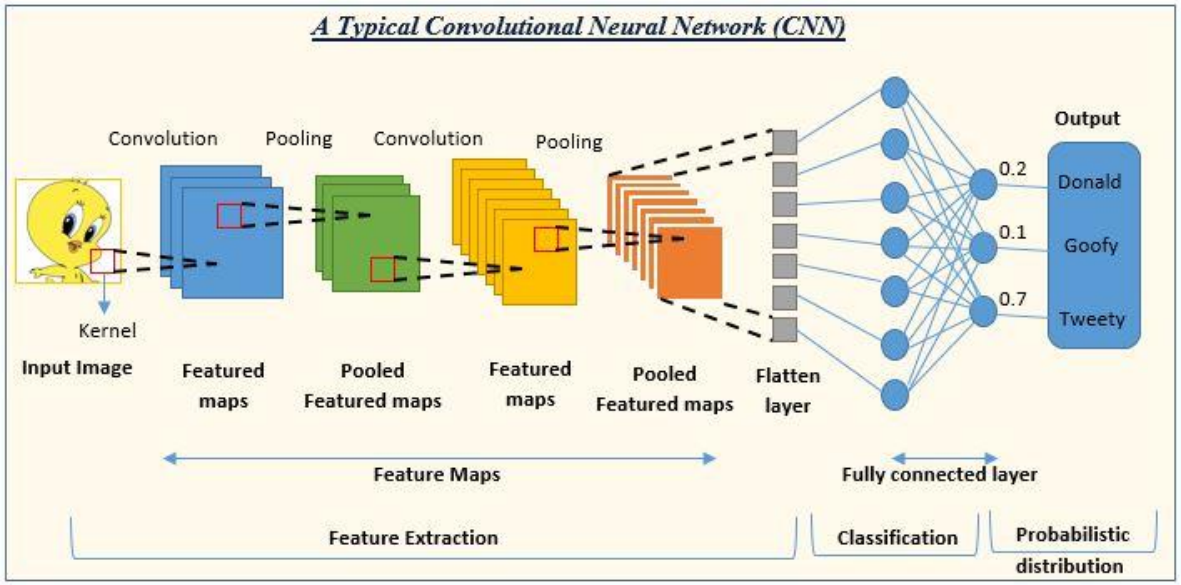
and other CPU providers

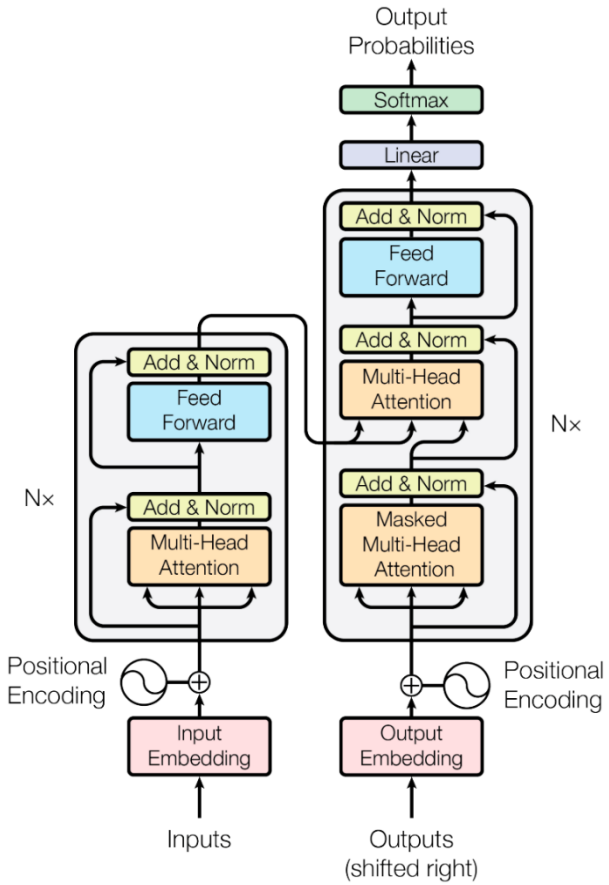
and other GPU providers

and other AI accelerator providers

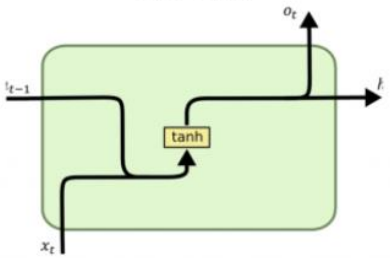
- Better models
  - Transformer
  - GNN
  - MCNN
- Deep Learning Framework improve
  - oneDNN
  - ZenDNN
  - IPEX
  - ONNX MLAS
  - .....
- Model Quantizer Tools
  - Intel Neural Compressor
  - TensorFlow Quantizer
  - PyTorch Quantizer

Major Neural Networks in Deep Learning

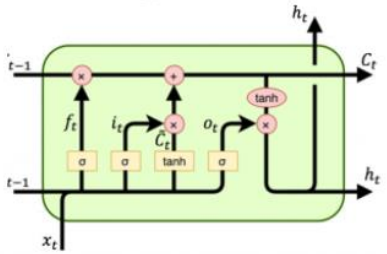




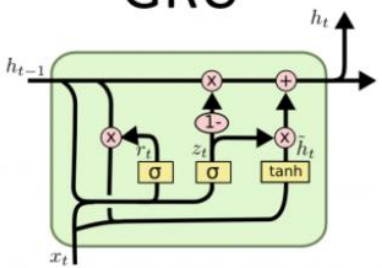
RNN



LSTM



GRU



TensorFlow PyTorch ONNX OpenVINO

http://dprogrammer.org/rnn-lstm-gru
https://baiblanc.github.io/2020/06/21/RNN-vs-CNN-vs-Transformer/
https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/

Major Network Security AI Usage Cases

- JavaScript malware detection
- Malware PE file detection
- Detect malicious C2 traffic
- Web/Email URL filtering
- IPS/IDS
- DLP
- .....

Intel® Network AI Offering

DISCOVERY	Data	Develop	Deploy
of possibilities & next steps	setup, ingestion & cleaning	models using analytics/AI	into production & iterate

Network AI

Simplify Network AI Deployment with Domain Expert Support (Use-case Ref.)

1. Design Use-case solution
2. Build AI models per use case
3. Optimize AI performance
4. E2E NW & AI Solution Deployment

One Intel AI foundation

Optimized Libraries, Frameworks, Tools

Developer Tools: BigDL, OpenVINO, oneAPI, Intel Neural Compressor, TADK

Standard Frameworks: TensorFlow, ONNX, PyTorch, scikit learn, LightGBM

Optimized Libraries:
- Intel® oneAPI Data Analytics Library (oneDAL)
- Intel® oneAPI Deep Neural Network Library (oneDNN)
- Intel® oneAPI Collective Communications Library (oneCCL)

End-to-End AI Portfolio Roadmap

Max Optimization on Intel® Processors: intel. XEON PLATINUM, intel. ATOM + intel. AGILEX, habana, Xe — Accelerate with Purpose

Intel® Deep Learning Boost (Intel® DL Boost), Intel® SSE4.1, Intel® Advanced Vector Extensions 512 (Intel® AVX-512), Intel® Advanced Matrix Extensions (Intel® AMX)

Intel® Xeon® Scalable Processors

DATA CENTER CPU OPTIMIZED FOR AI

INTEL® ADVANCED VECTOR EXTENSIONS 512 (INTEL® AVX-512)
INTEL® DEEP LEARNING BOOST (INTEL® DL BOOST)
INTEL® OPTANE™ DC PERSISTENT MEMORY

Intel® DL Boost Technologies			
Microarchitecture	AVX512_VNNI	AVX512_BF16	AMX
Client			
10th Gen Intel Core	✓	X	X
Server			
2nd Gen Intel Xeon SP	✓	X	X
3rd Gen Intel Xeon SP	✓	✓	X
3rd Gen Intel Xeon SP	✓	X	X
4th Gen Intel Xeon SP	✓	✓	✓

2019	2020	2021	2022+

3RD GEN INTEL XEON SP

Intel® Deep Learning Boost
Intel® AVX-512 (VNNI/INT8 & BFloat16)

4TH GEN INTEL XEON SP

Intel® AMX – INT8 and BFloat16 support
Intel® AVX-512 (VNNI/INT8)

2ND GEN INTEL XEON SP

Intel® Deep Learning Boost (Intro)
Intel® AVX-512 (VNNI/INT8)

3RD GEN INTEL XEON SP

Intel® Deep Learning Boost
Intel® AVX-512 (VNNI/INT8)

LEADERSHIP PERFORMANCE

Intel® oneAPI Deep Neural Network Library (oneDNN)

Features

- Supports FP32, FP16, Bfloat16, and int8.
- Leverages Intel® DL Boost, Intel® AVX-512 instructions, and processor capabilities
- Fused operations for optimized performance

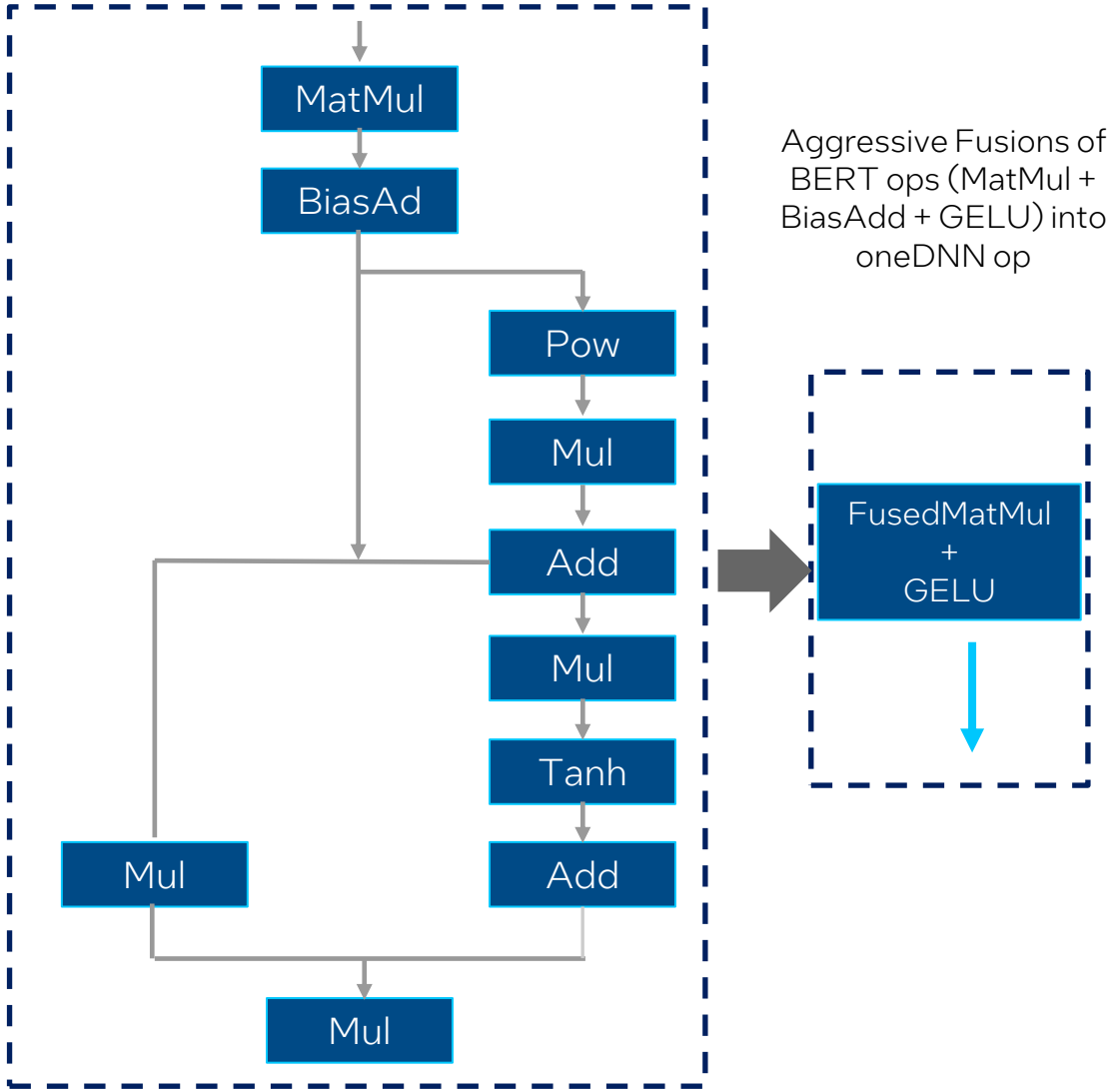
Support Matrix

- Compilers: Intel® oneAPI DPC++ / C++ Compilers
- OS: Linux, Windows, macOS
- CPU: Intel Atom®, Intel® Core™, Intel® Xeon®, Intel® Xeon® Scalable processors
- GPU: Intel® Processor Graphics Gen9, Intel® Processor Graphics Gen 12

Category	Functions
Compute intensive operations	• (De-)Convolution • Inner Product • RNN (Vanilla, LSTM, GRU) • GEMM
Memory bandwidth limited operations	• Pooling • Batch Normalization • Local Response Normalization • Layer Normalization • Elementwise • Binary elementwise • Softmax • Sum • Concat • Shuffle
Data manipulation	• Reorder

Intel® oneDNN Integration with TensorFlow

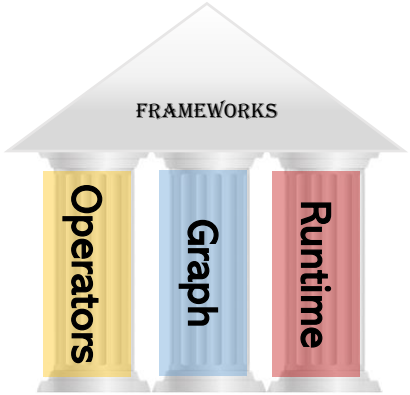
- Replaces compute-intensive standard TF ops with highly optimized custom oneDNN ops
- Aggressive op fusions to improve performance of Convolutions and Matrix Multiplications
- Primitive caching to reduce the overhead of calling oneDNN
- oneDNN accelerates inference performance with automatic detection of Intel® Deep Learning Boost technology and the latest supported CPU instruction sets. That means the users can get significant performance improvement under Intel oneDNN enhanced deep learning framework without changing their codes.
- oneDNN optimizations at runtime in official TensorFlow distributions and has been enabled by default since TensorFlow 2.9



https://github.com/tensorflow/community/pull/400
https://medium.com/intel-analytics-software/leverage-intel-deep-learning-optimizations-in-tensorflow-129faa80ee07

Intel® Optimization for PyTorch

3-Pillar Framework Optimization Techniques



Operators

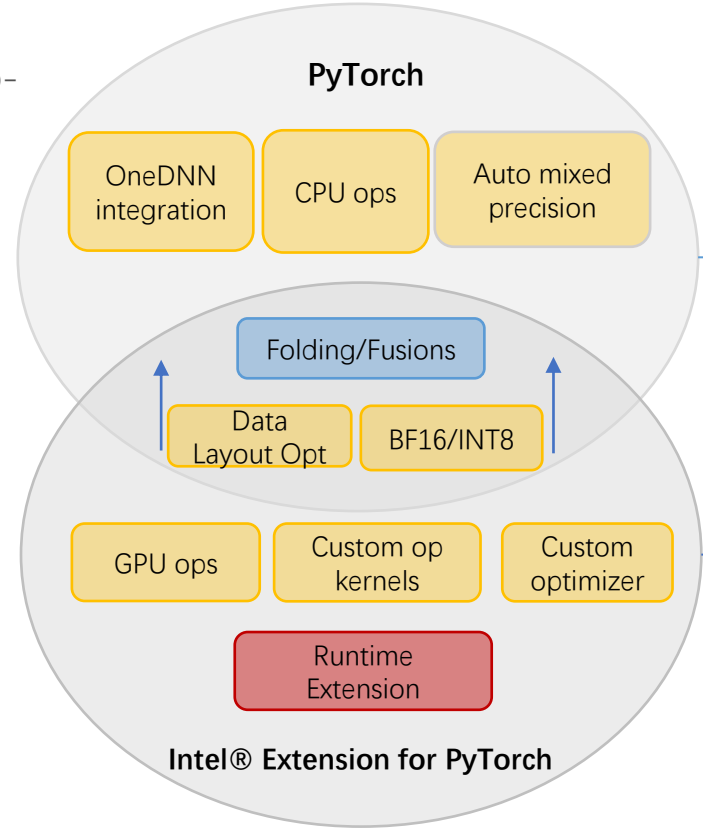
- ✓ Vectorization and Multi-threading
- ✓ Low-precision BF16/INT8 compute
- ✓ Ease-of-use BF16 compute with Auto-Mixed-Precision (AMP)
- ✓ Data layout optimization for better cache locality

Graph

- ✓ Constant folding to reduce compute
- ✓ Op fusion for better cache locality

Runtime

- ✓ Thread affinity and multi-streams
- ✓ Memory buffer pooling
- ✓ GPU runtime
- ✓ Launcher

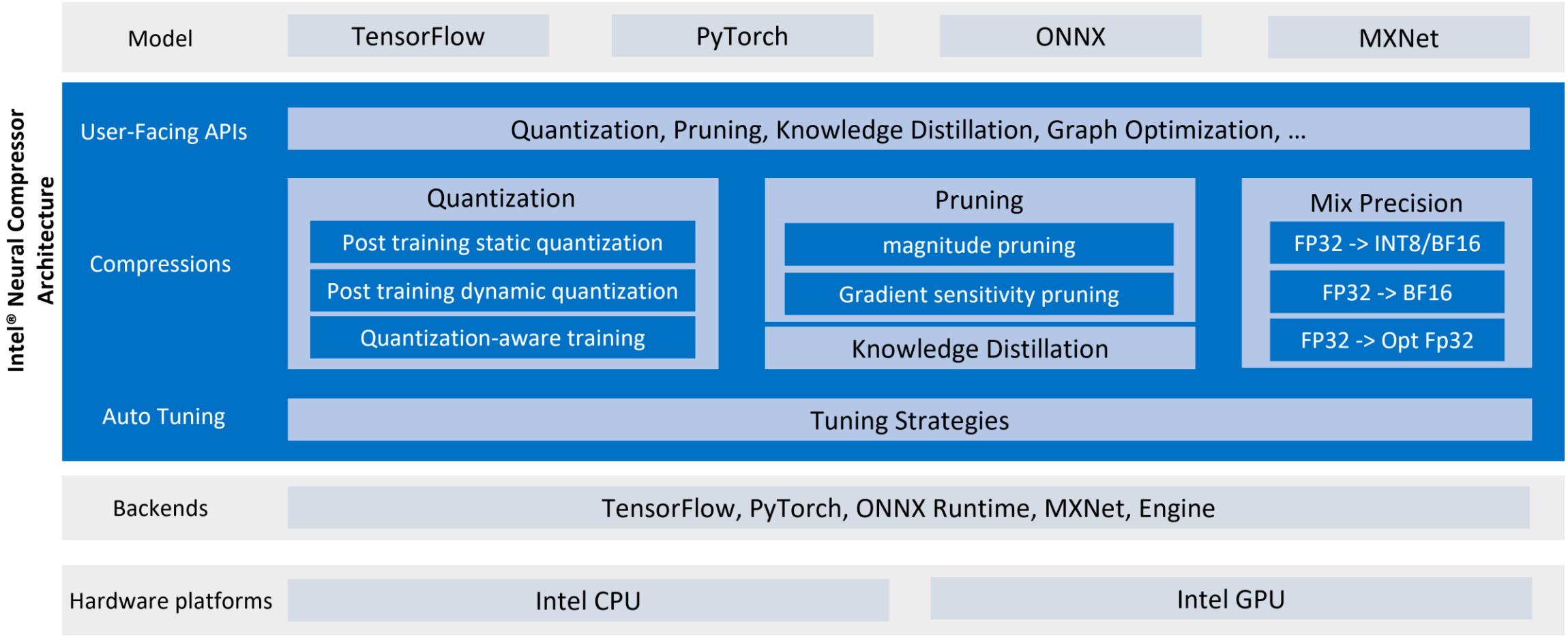


PyTorch Upstream
General Intel new feature enabling and performance optimizations

Intel Extension for PyTorch
Early access/adoption of aggressive optimizations & GPU support

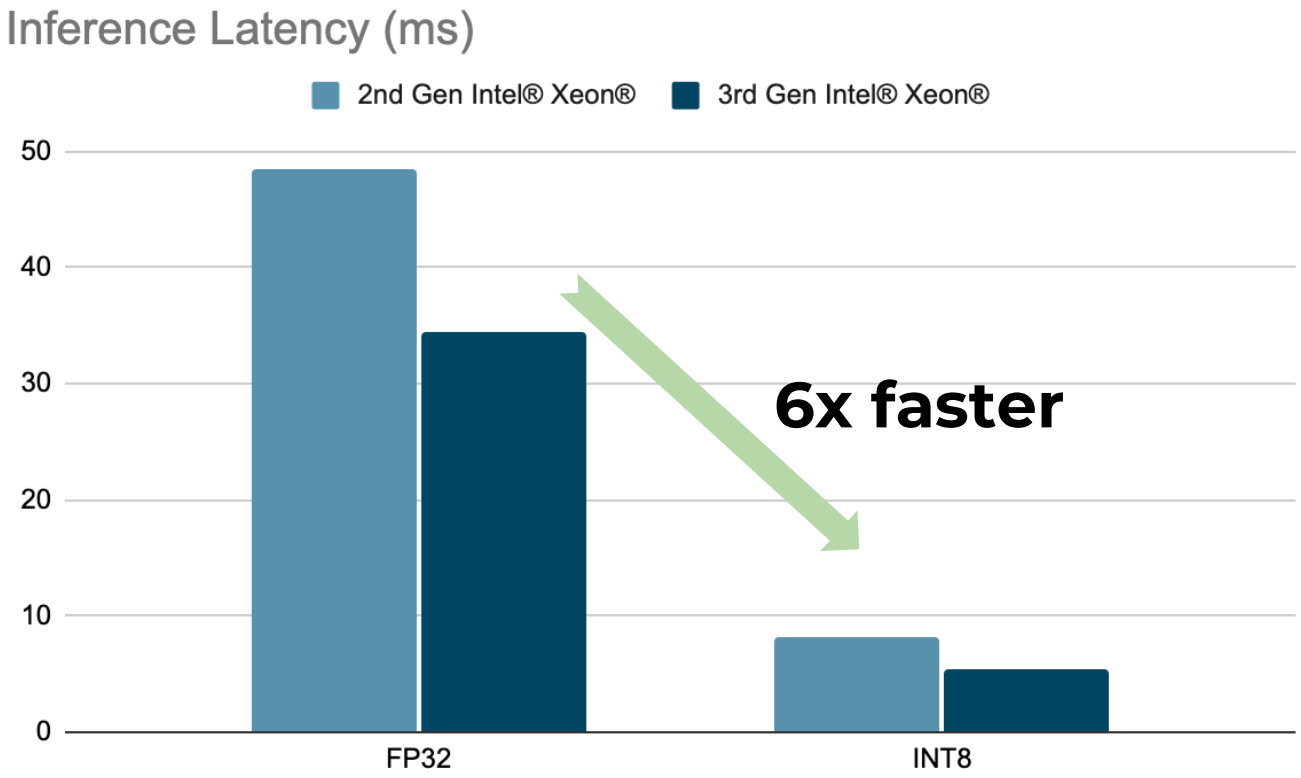
Intel® Neural Compressor Infrastructure

Opensource Tool for quantization(https://github.com/intel/neural-compressor)



AI Inference Latency Boost

Palo Alto Networks New Advanced Threat Prevention - Industry's first IPS to stop unknown C2

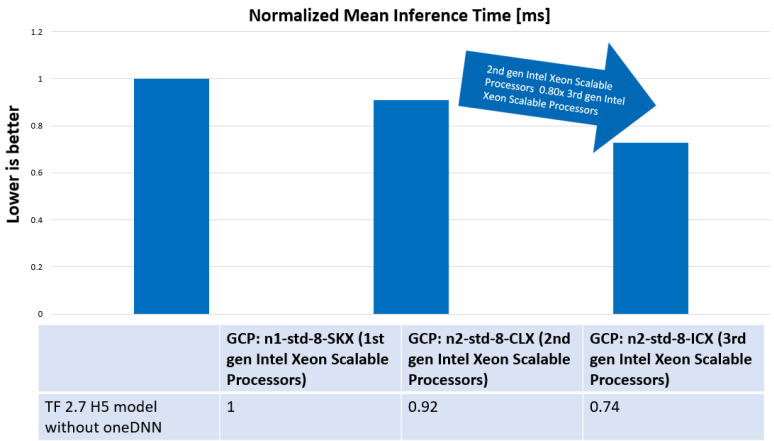


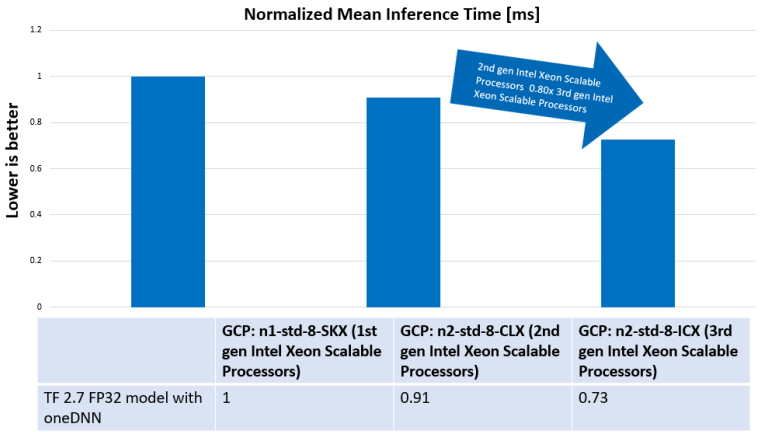
- Tests were done on GCP N2 instances with 8-VCPU 2nd Gen Intel Xeon or 3rd Gen Intel Xeon processor
- INT8 model is ~6x faster than original SavedModel
- 3rd Gen Intel Xeon is ~30% faster than 2nd Gen Intel Xeon processor

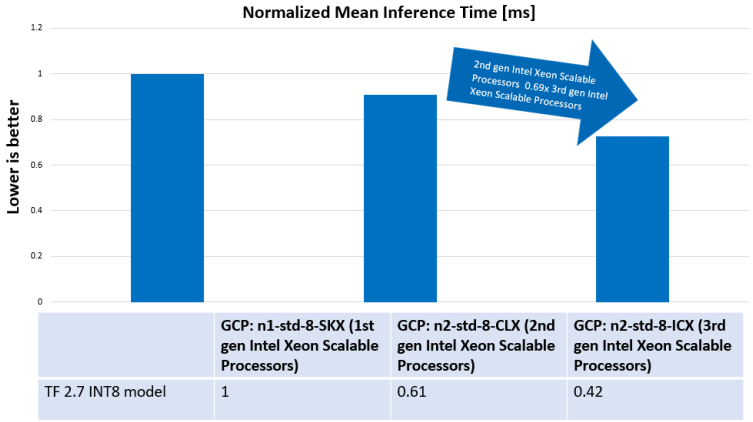
https://networkbuilders.intel.com/social-hub/webcast/next-gen-security-services-with-inline-ai-inference

CNN Model: EMBER model deep dive and optimization

Detect novel, unknown malware in Executable Files







Inference time(ms) one core	TensorFlow 2.7				Performance Boost H5 model vs INT8 model
	H5 model W/O oneDNN	FP32 model W/O oneDNN	FP32 model with oneDNN	INT8 model	
GCP: n1-std-8-SKX((1st gen Intel Xeon Scalable @2.00GHz)	**121.56**	78.63	45.54	46.49	**2.61 X**
n2-std-8-CLX (2nd gen Intel Xeon Scalable @2.80GHz)	112.22	69.21	41.4	28.15	**3.99 X**
n2-std-8 (3rd gen Intel Xeon Scalable @2.60GHz)	90.34	53.4	33.08	19.47	**4.64 X**

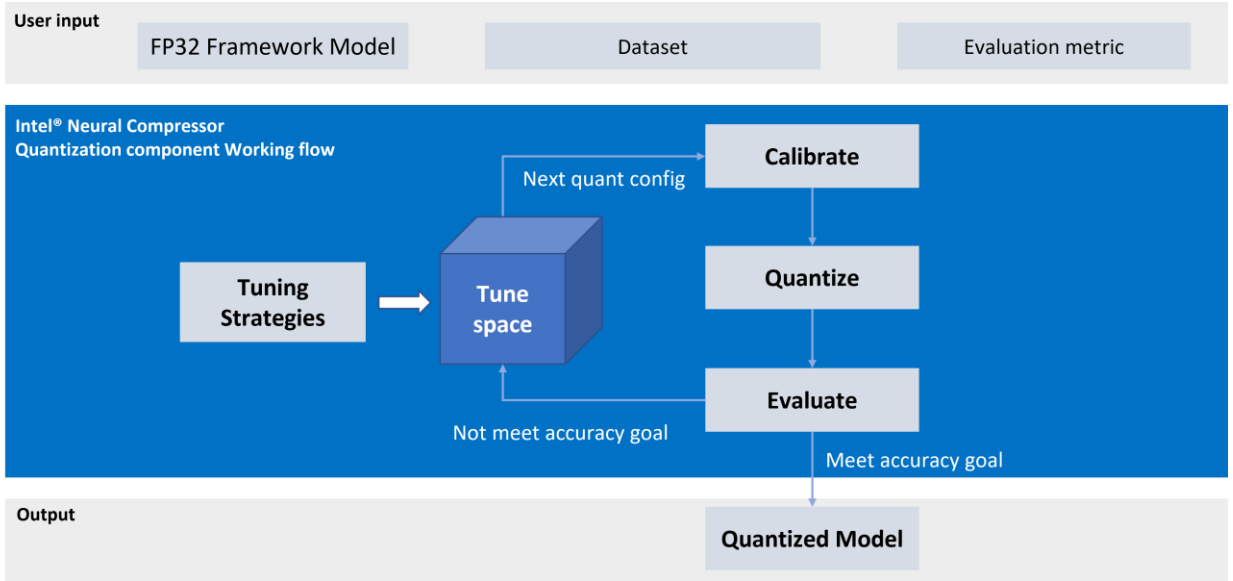
https://networkbuilders.intel.com/solutionslibrary/intel-deep-learning-boost-boost-network-security-ai-inference-performance-in-google-cloud-platform-gcp-technology-guide

Examples: Quantize TensorFlow RN50

```
# RN50.yaml
model:
  name: resnet50
  framework: tensorflow

quantization:
  calibration:
    dataloader:
      dataset:
        ImageRecord:
          root: /path/to/calibration/dataset
      transform:
        ResizeCropImagenet:
          height: 224
          width: 224
          mean_value: [123.68, 116.78, 103.94]

evaluation:
  accuracy:
    metric:
        topk: 1
```




```
# tune.py
from neural_compressor.experimental import Quantization, common

quantize = Quantization('./RN50.yaml')
quantize.model = common.Model(self.args.input_graph)
q_model = quantize()
q_model.save(output_model_path)
```

Summary

- How to improve the performance with Intel® oneDNN and Intel® Neural Compressor under TensorFlow: https://networkbuilders.intel.com/solutionslibrary/intel-deep-learning-boost-boost-network-security-ai-inference-performance-in-google-cloud-platform-gcp-technology-guide
- Next-Gen Security Services with Inline AI Inference: https://networkbuilders.intel.com/social-hub/webcast/next-gen-security-services-with-inline-ai-inference
- AI Technologies – Unleash AI Innovation in Network Applications: https://networkbuilders.intel.com/solutionslibrary/ai-technologies-unleash-ai-innovation-in-network-applications-solution-brief

intel®