3 Lessons Learned in Building a Successful AI Inferencing Toolkit

Yury Gorbachev
OpenVINO Architect, Senior Principal Engineer
Intel

VISION PRODUCT OF THE YEAR 2019

2019 Developer Tool of the Year
Awarded by the Edge AI and Vision Alliance

OpenVINO™

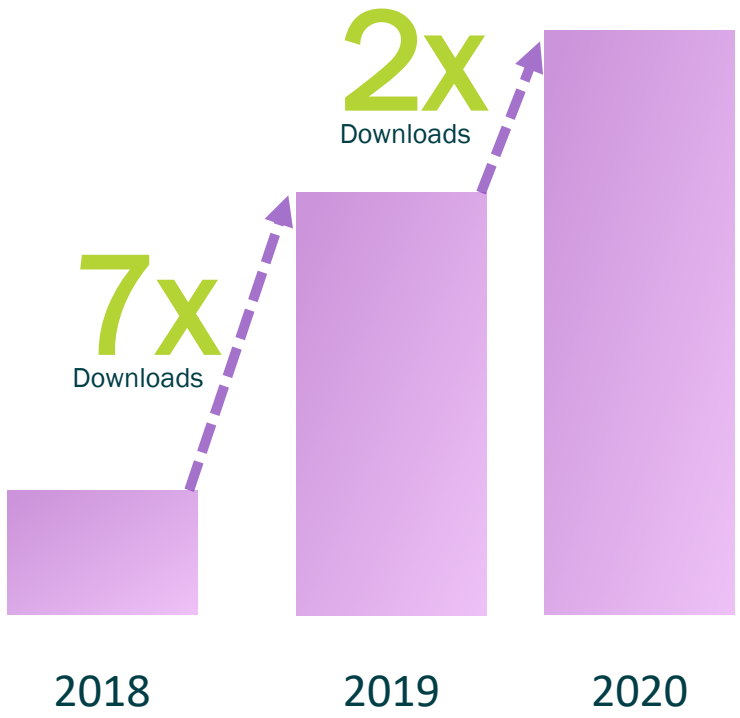
What is the OpenVINO toolkit?

- Set of products to assist in development of AI powered applications
  - Inference libraries as base offering, additional components to help
  - Efficient deployment (power and performance) is goal #1
- Support for Intel AI family and other architectures
  - ARM support available within open-source version
- 3 years in full production mode
  - Pipeline of commercial customers
  - Developed in open source, stable quarterly releases, LTS release every year

Why do we think we are successful?

2021 embedded VISION summit

Developer downloads since global launch in 2018



Proud to be working with our customers

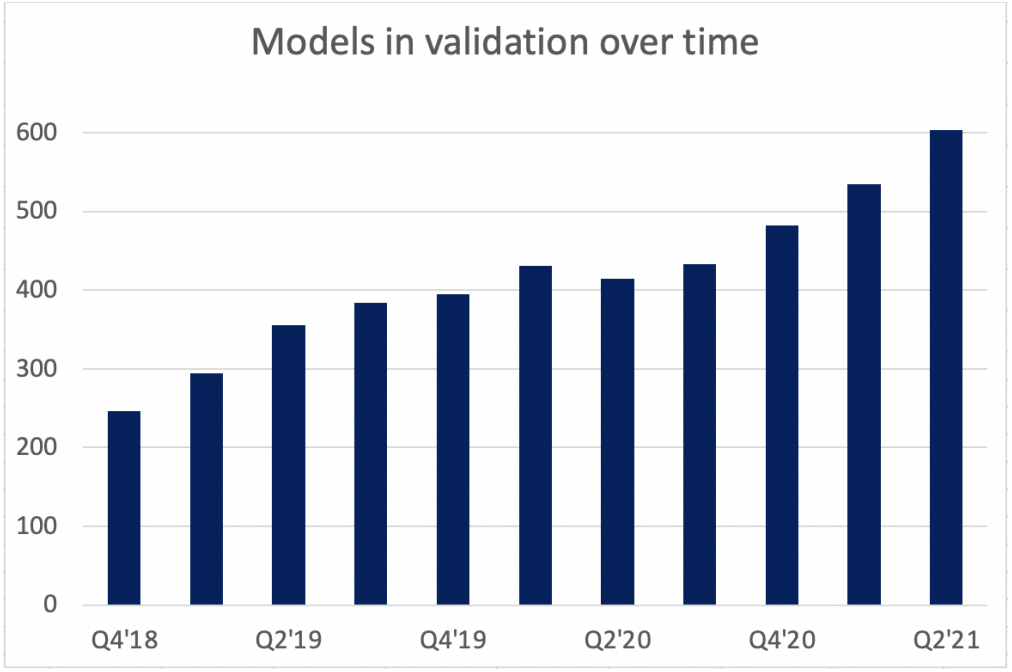
OpenVINO™

Prepare solution for out of box scalability

- Evolution in the AI field is explosive
  - New architectures being produced (e.g. Inception v4: introduced in 2016 - rarely used in new production now)
  - Constant expansion of solved problems (classification, detection, action recognition, translation, ...)
  - AutoML domain leads to exponential growth of potential model architectures that customers run
- Tuning solution for small set of models will simply not work
  - Both when creating SW and HW
  - Also: benchmarks like MLPerf are not an indicator of SW/HW maturity for customer tasks
- Having robust and scalable product is a must
  - When a customer reports a problem – you are already late with your solution
  - Scalable generic components are beneficial (graph compilers, automatic fusions, etc.)
  - Ability to expand functionality quick (add/change ops)

How we are solving the “explosive AI” problem

- Ensure we stay up to date with latest trends in research
  - Gap between research and production adoption is usually few quarters
  - Collaborating with data science teams (we have a few) to stay updated
  - Some emerging solutions are getting adopted within our Model Zoo offering as well
- Established and mature validation/design environment
  - Hundreds of models in regular validation (growing!)
  - Analyzing execution hotspots, memory consumption, etc.
  - Layer parameter and subgraph extraction
  - Automated performance and accuracy validation on datasets



Few roadblocks to deployment tool adoption

- Not all customers have established culture of deploying models
  - Approach to move from development to deployment is not well systemized (industrywide)
  - No widespread understanding of advantages for separate runtime
  - Deployment with framework is not rare – it is simplest, and adoption of separate toolkit is not easy
- Opposite situation – models evolving frequently and becoming living artefact
  - Continuous model evolution via new data, rapid path to production with updated model
  - No place for manual modifications or any non-automated actions

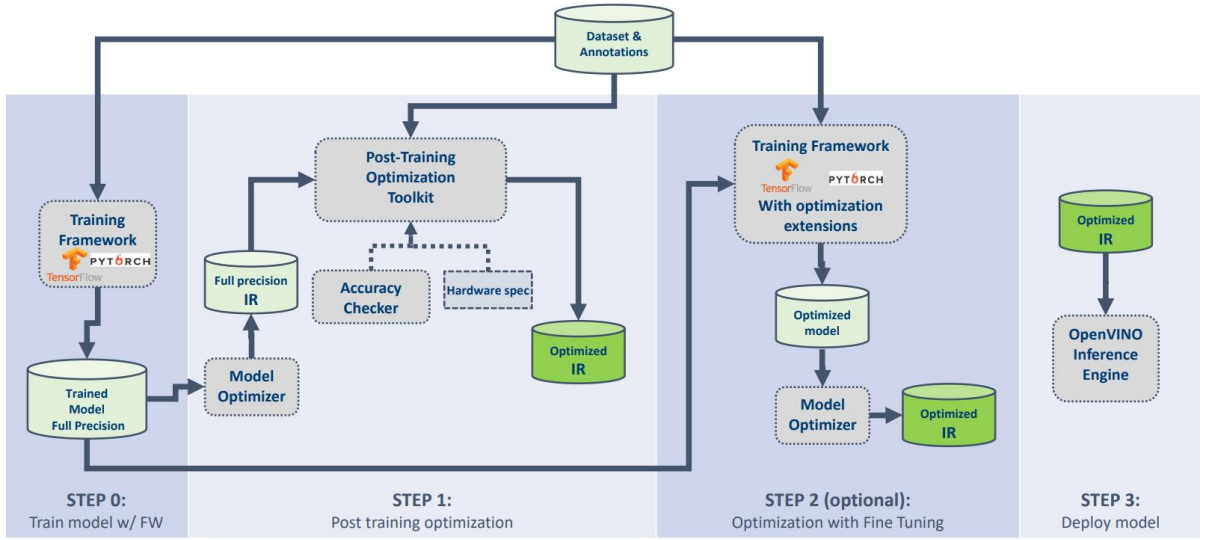
- Out of the box conversion and optimization of models is critical
- Being backward compatible on tools level is important

How we solve the “out of the box” problem

- Leverage large set of reference applications
  - Execute network != Execute application
    - Audio workloads require states within networks to work correctly
    - NLP is operating on dynamic inputs and requires dynamism support
  - Learn domain specifics and incorporate it in APIs (evolving is hard!)
- Substantial investment in ensuring backward compatibility
  - Decided on use cases to support: Run old models, old applications, sustain performance, etc.
  - Carefully covered in architecture reviews, scenario analysis, model/operation versioning
  - Backed by strong functional validation, design level tests
- Not yet fully solved: Simple migration from framework for arbitrary models

Ecosystem offering has higher customer adoption

- We started with single inference runtime solution
- Evolved into ecosystem of connected products
  - Pretrained models, network optimization tools, video pipeline frameworks
  - Performance analysis and visualization
- Combined offering provides more flexibility
  - Optimize for specific HW & runtime
  - Offload vision pipelines to accelerator
  - Produce most efficient models for dataset
- Exponentially more complicated



Low precision optimization flow within OpenVINO

How we build the ecosystem

- Mostly looking at customer demands and unsolved challenges on path to deployment
  - Simplifying developer flow is our main goal, fewer environment changes
  - Fewer dependencies is a plus. Not all of them possible to adapt to your demands
  - Not all ways of solving problems are easy for deployment-focused engineers
    - E.q. post-training network quantization is simpler than fine-tuning
- A lot of interconnections between different tools
  - Download public model from model zoo, quantize for target, measure accuracy
  - Train model template on own dataset, optimize via fine-tuning and export to OpenVINO
- Large educational investment
  - DL Workbench design tool with Jupyter Notebooks, samples, Model Zoo with public models

Conclusions: what can help you to progress faster?

- Collaborate with data science teams to learn and evolve in different domains
  - Become aware of state of the art networks and usage in applications
- Invest in robust and scalable validation environment
  - Can take weeks for full cycle, enormous time invesment, full automation is a must
- Reduce developer friction, build ecosystem of tools
- Invest in educating your developers
  - Most of them are not deep learning/AI savvy

3 Lessons Learned in Building a Successful AI Inferencing Toolkit

3 Lessons Learned in Building a Successful AI Inferencing Toolkit

1. Solve the “Exploding AI” problem
2. Innovate over “out of the box” challenge
3. Invest into ecosystem and education

Learn More

OpenVINO toolkit (Documentation):
https://docs.openvinotoolkit.org/

GitHub for all OpenVINO products (open source):
https://github.com/openvinotoolkit

2021 Embedded Vision Summit

Join us for more sessions on Intel® Distribution of OpenVINO™ toolkit!

OVER-THE-SHOULDER TUTORIAL

The Fearless AI Challenge: Can You Quickly Deploy AI Inference to Billions of Devices?

PANEL DISCUSSIONS

Why is Taking AI to Production so Difficult?

EXPERT BAR

Learn How You Can Accelerate AI Inference and Containerized Workloads at the Edge

OpenVINO™

Notices and Disclaimers

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

Thank you!

OpenVINO™