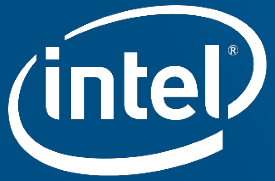


ACCELERATING MACHINE LEARNING SOFTWARE ON IA

Ananth Sankaranarayanan
Director of Engineering, Analytics & AI Solutions Group, Intel

November 2016

Abstract

This session reviews Intel's hardware and software strategy for machine learning. Intel is offering a range of tools and partnering with the open source community to help developers deliver optimized machine learning applications for Intel Architecture systems. This session spans a range of software development frameworks, libraries and other tools for machine learning with a focus on performance. Topics include IA-optimized frameworks such as Apache Spark*, Berkeley Caffe*, Google Tensorflow, and Nervana Neon and related performance benchmarks on the latest Intel Xeon and Intel Xeon Phi™ processors. Machine learning libraries Intel® Math Kernel Library and Intel® Data Analytics Acceleration Library will also be highlighted along with the new Intel® Distribution for Python and Intel® Deep Learning SDK.

Speaker Bio

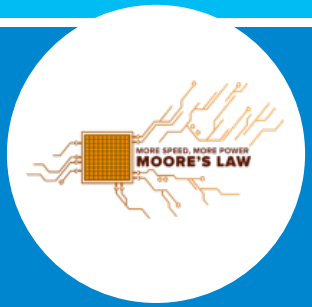
Ananth Sankaranarayanan is currently the Director of Engineering with Intel in the Analytics and Artificial Intelligence Solutions group. His team is responsible for performance, partner and solution engineering functions, driving new platform initiatives jointly with leading Cloud Service Providers, Hardware Manufacturers and Software Vendors worldwide delivering engineered solutions to simplify implementations. Ananth previously led the HPC program for Intel silicon design/manufacturing and delivered 5 successful generations of supercomputers that directly contributed to reducing Intel Silicon Time to Market, and he has been with Intel since 2001. Ananth received his bachelors in computer science and engineering from Bharathidasan University in India and his Masters in Business Administration from City University of Seattle, USA.

Drivers for fast emergence of AI

Bigger Data

Image: 1 MB / picture
Audio: 5 MB / song
Video: 5 GB / movie

Better Hardware



Significant compute performance increases year over year

Parallel processing norm now

Smarter Algorithms

Advances in algorithm innovation, including neural networks, leading to better accuracy in training models

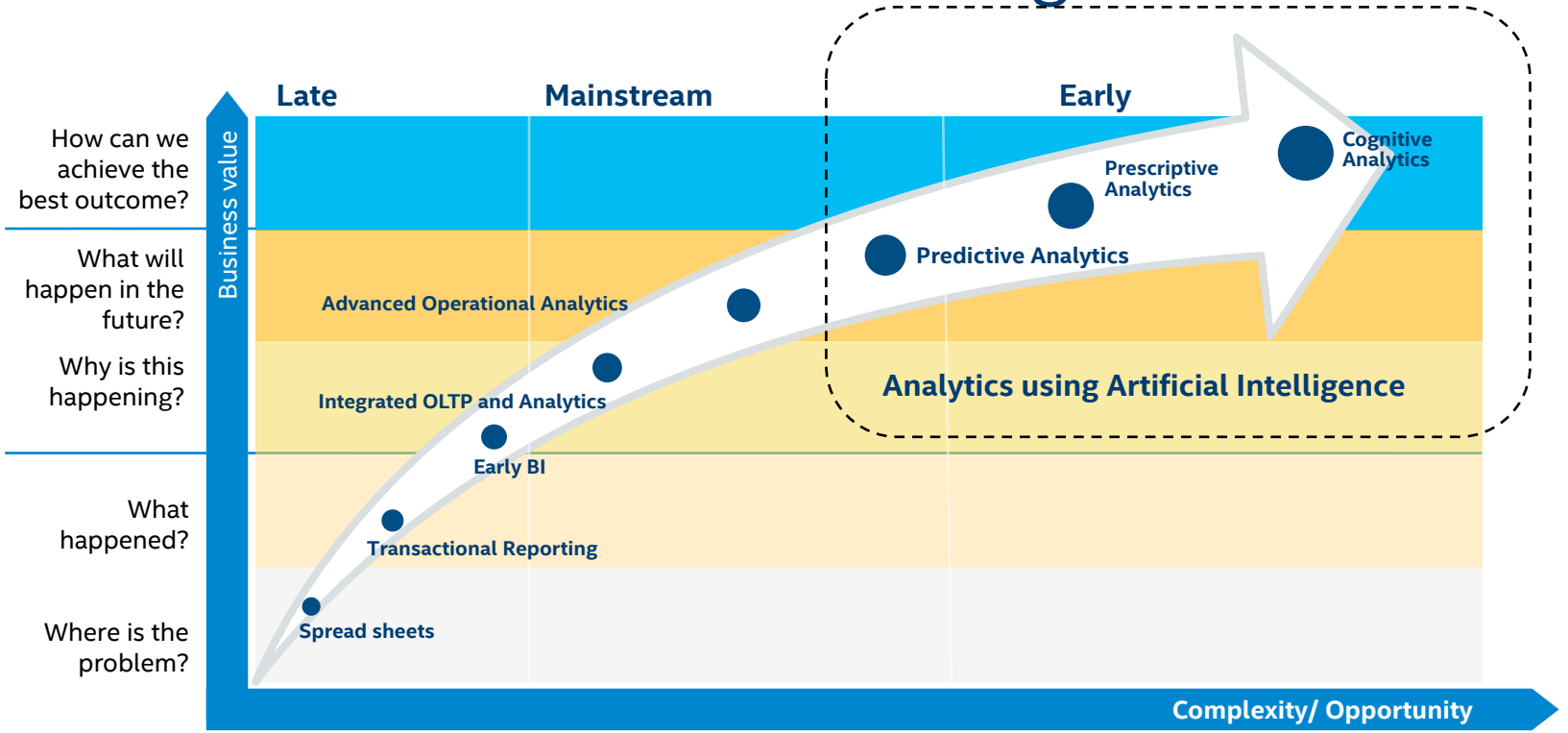
Data + Analytics Creates Unique Opportunities

Companies that use analytics best are...

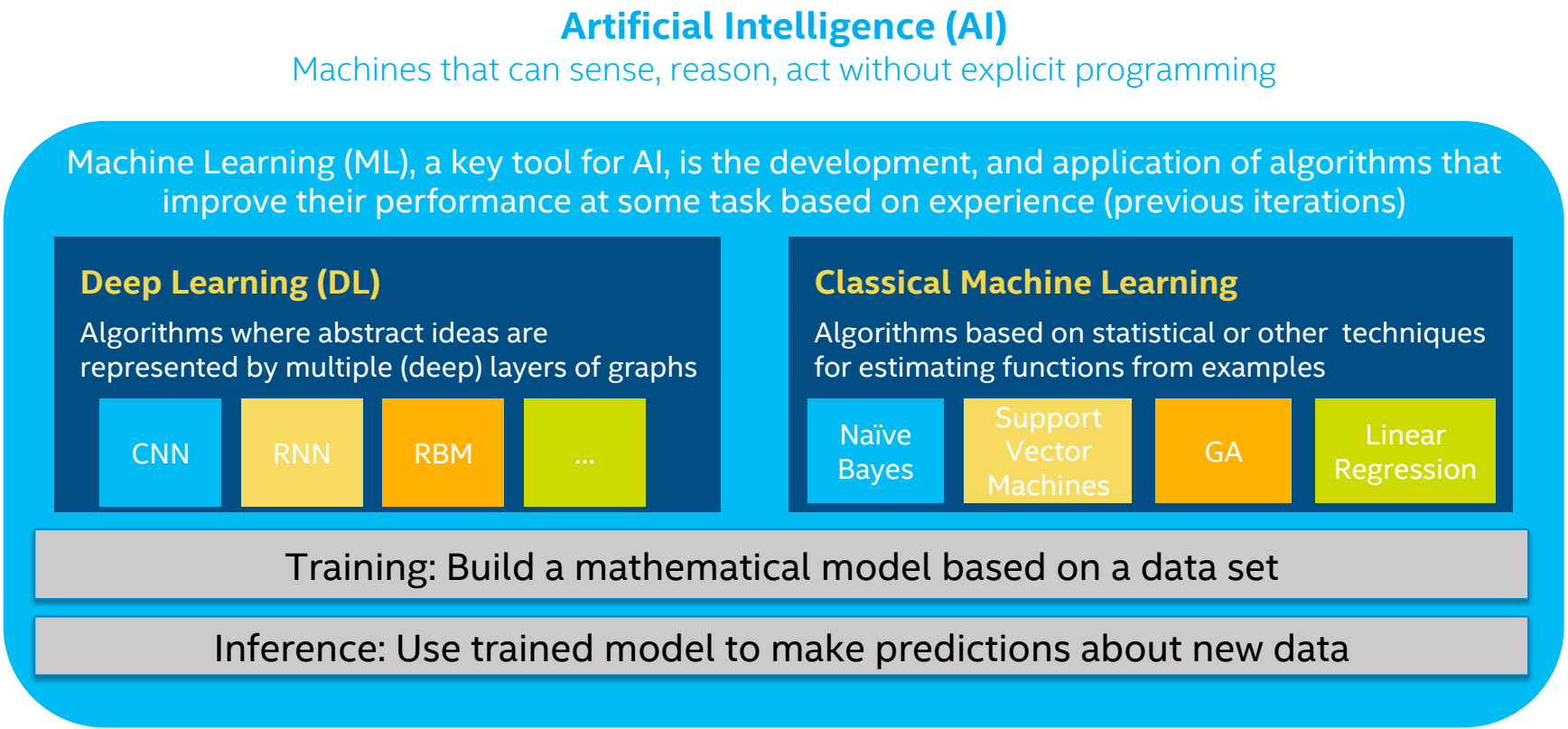
2X	5X	3X	2X
...more likely to			
Make data-driven decisions	Make decisions faster than others	Execute on decisions faster	Have top-quartile financial results

Source: Bain

The Evolution of Artificial Intelligence



Taxonomy


Artificial Intelligence (AI)
Machines that can sense, reason, act without explicit programming
Machine Learning (ML), a key tool for AI, is the development, and application of algorithms that improve their performance at some task based on experience (previous iterations)
Deep Learning (DL)
Algorithms where abstract ideas are represented by multiple (deep) layers of graphs
CNN
RNN
RBM
...
Classical Machine Learning
Algorithms based on statistical or other techniques for estimating functions from examples
Naïve Bayes
Support Vector Machines
GA
Linear Regression
Training: Build a mathematical model based on a data set
Inference: Use trained model to make predictions about new data

AI: Deep Learning Example:

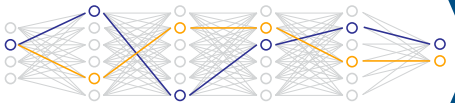
Step 1: Training

(In Data Center – Over Hours/Days/Weeks)

Massive data sets: labeled or tagged input data

Person

Create “Deep neural net” math model



Trained Model

Output Classification

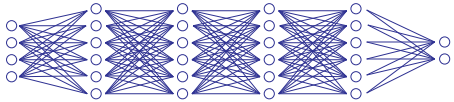
90% person
8% traffic light

Step 2: Inference

(At the Edge or in the Data Center - Instantaneous)

New input from camera and sensors

Trained neural network model



97% person

Output Classification

AI: Example Use Cases

Cloud Service Providers

Financial Services

Healthcare

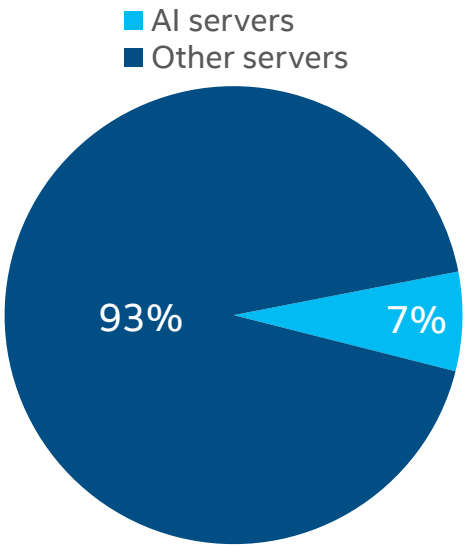
Automotive

- Image classification and detection for accurate diagnosis
- Image recognition/ tagging for defect identification
- Natural language recognition (digital assistants)
- Big data pattern detection
- Targeted ads to increase revenue
- Fraud prevention/ face detection
- Gaming, check processing
- Computer server monitoring
- Safe navigation for autonomous vehicles
- Financial forecasting and prediction to avoid risk
- Network intrusion detection

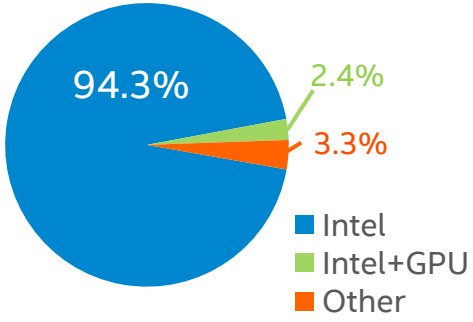
AI Market Opportunity

PRESENT → FUTURE

Server Market (2015)[1]

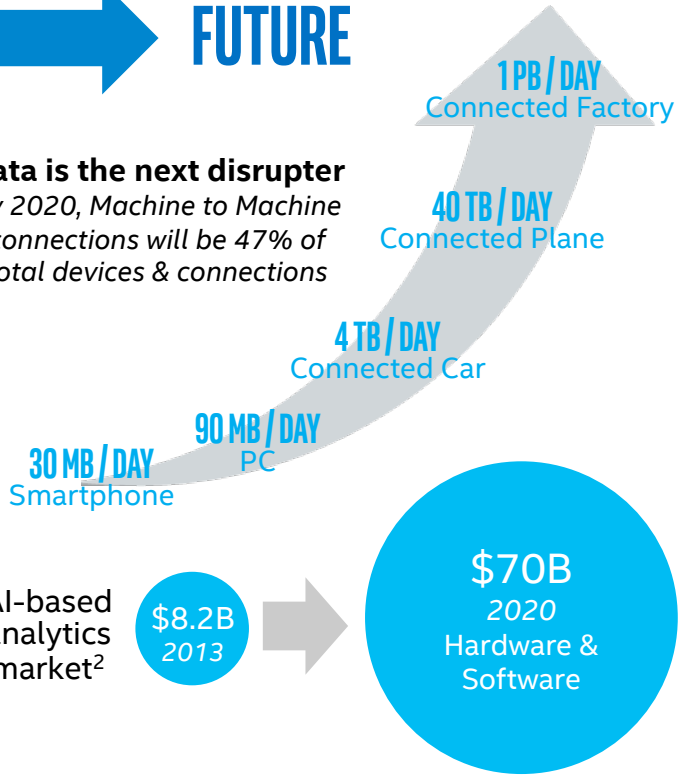


Architecture MSS



Data is the next disrupter

By 2020, Machine to Machine connections will be 47% of total devices & connections



[1]Source: DCG Market Intelligence team

[2]Source: IDC, IOT market related to analytics

INTEL AI STRATEGY

Intel AI strategy

Making AI more pervasive by enabling deployment ready AI solutions through a large, open ecosystem

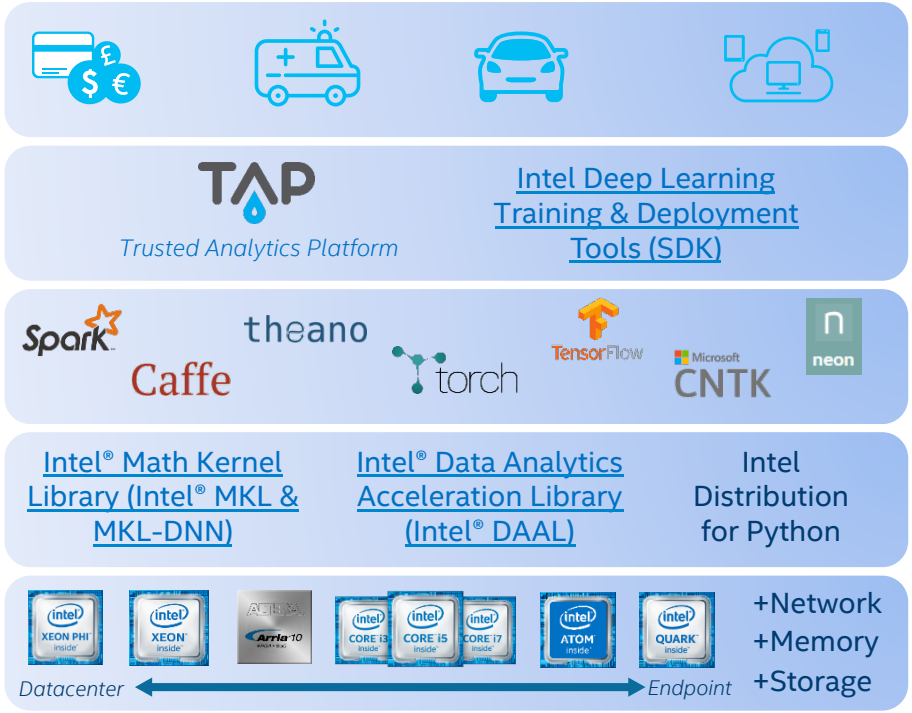
Solution blueprints
for reference across industries

Tools/Platforms
to accelerate deployment of IA solution stack

Optimized Open Frameworks
that scale to multi-node and deliver best performance

Free Libraries/Languages
featuring optimized ML/DL building blocks to enable developers

Best in class hardware
Cross compatible portfolio spanning from data center to edge delivering high perf, perf/TCO, perf/w



AI: HARDWARE

Intel AI Products for the Datacenter

Training

Intel® Xeon Phi™ Processors

- Optimized for performance
- Scales with cluster size for shorter time to model
- x86 architecture, consistent programming model for training and inference

Inference

Intel® Xeon® Processors

- Optimized for performance/TCO
- Most widely deployed inference solution

Intel® Xeon® Processors + FPGA (discrete)

- Optimized for performance/watt
- Reconfigurable – can be used to accelerate many DC workloads
- Programmable with industry standard OpenCL

Intel® Xeon Phi™ Processor Family

Enables shorter time to train

Breakthrough Highly-Parallel Performance

- Up to ~6 SGEMM TFLOPs[3] per socket
- Great scaling efficiency resulting in lower time to train for multi-node
- Eliminates add-in card PCIe* offload bottleneck and utilization constraints

Removes Barriers through Integration

- Integrated Intel® Omni-Path fabric (dual-port; 50 GB/s) increases price-performance and reduces communication latency for deep learning networks

Better Programmability

- Binary-compatible with Intel® Xeon® processors
- Open standards, libraries and frameworks

Configurations: 1-8 see page 16.

All specifications refer to the future Intel® Xeon Phi™ processor (Knights Landing) unless otherwise noted
All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/performance *Other brands and names are the property of others.

Mainstream Training: Intel® Xeon Phi™ Processor 7250

Competitive Deep Learning Image Classification SINGLE-NODE Training With Mainstream cuDNN

Topology: **Caffe*/AlexNet[1]**

Dataset: **Large image database**

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance . *Other names and brands may be property of others

Configurations:

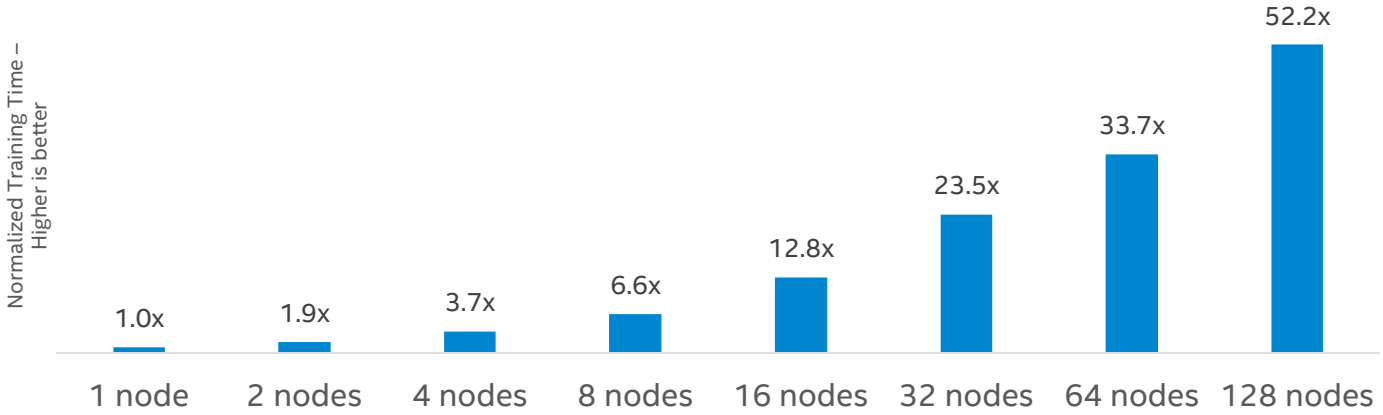
1. Nvidia Tesla M40* (core@923MHz, 12GB, mem@3004MHz, 250W), DIGITS Deep Learning Machine* hosted on 2S Intel® Xeon® processor E5-2620 v3, 64GB memory, Ubuntu* 14.04, Nvidia* Driver v352.41, cuDNN v4, BVLC/Caffe cuDNN v5 or NVIDIA/Caffe cuDNN v5
2. Intel® Xeon Phi™ processor 7250 (68 Cores, 1.4 GHz, 16GB), 128GB memory, Red Hat* Enterprise Linux 6.7, Intel® Caffe: https://github.com/intelcaffe

AlexNet: https://papers.nips.cc/paper/4824-Large image database-classification-with-deep-convolutional-neural-networks.pdf, Batch Size:256; ** Q4'16 is estimated based on MKL engineering version

Why does Scaling Matter?

Train Up to 50x faster with Intel® Xeon Phi™ Processor

Deep Learning Image Classification Training Performance - MULTI-NODE Scaling



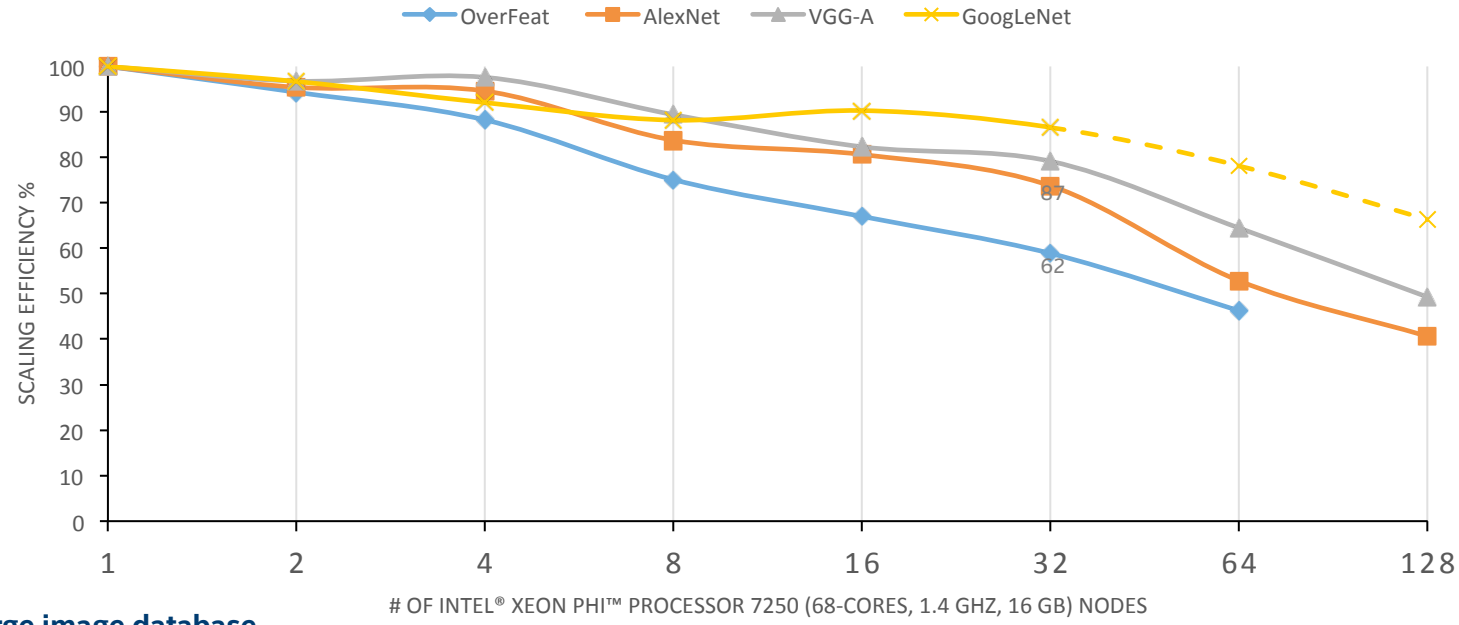
Topology: **AlexNet***

Dataset: **Large image database**

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance/datacenter. Configurations: Up to 50x faster training on 128-node as compared to single-node based on AlexNet* topology workload (batch size = 1024) training time using a large image database running one node Intel Xeon Phi processor 7250 (16 GB MCDRAM, 1.4 GHz, 68 Cores) in Intel® Server System LADMP2312KXXX41, 96GB DDR4-2400 MHz, quad cluster mode, MCDRAM flat memory mode, Red Hat Enterprise Linux* 6.7 (Santiago), 1.0 TB SATA drive WD1003FZEX-00MK2A0 System Disk, running Intel® Optimized DNN Framework, training in 39.17 hours compared to 128-node identically configured with Intel® Omni-Path Host Fabric Interface Adapter 100 Series 1 Port PCIe x16 connectors training in 0.75 hours. Contact your Intel representative for more information on how to obtain the binary. For information on workload, see https://papers.nips.cc/paper/4824-Large image database-classification-with-deep-convolutional-neural-networks.pdf.

Great Scaling Efficiency: Intel® Xeon Phi™ Processor

Deep Learning Image Classification Training Performance - MULTI-NODE Scaling

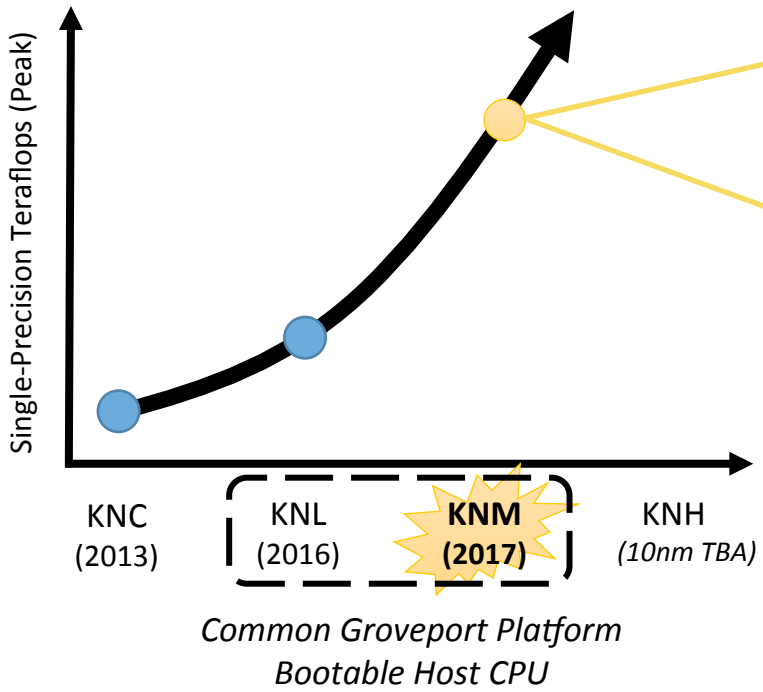


Dataset: **Large image database**

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance . *Other names and brands may be property of others
Configurations: Intel® Xeon Phi™ Processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM), 128 GB memory, Red Hat* Enterprise Linux 6.7, Intel® Optimized Frameworks

Knights Mill: Next Gen Intel® Xeon Phi™ processor

Enables shorter time to train



Trains Machines Faster

- >2X* Single Precision & >4X* 16-bit Mixed Precision faster deep learning training performance
- Highly distributed processing with efficient scaling over multi-node offers flexible infrastructure for ML/DL workloads

Consistent Programming Model

- Common Xeon & Xeon Phi programming for developers
- Optimized for industry standard Open Source frameworks
- Bootable Host-CPU avoids offloading latency & bottleneck

Memory Flexibility

- High memory bandwidth with integrated DRAM increases performance for complex neural datasets by reducing latency
- Large DDR4 memory capacity for massive AI use cases

*NOTE: Performance theoretical wrt KNL7250 SKU based on KNM architectural changes.

Intel® Xeon® Processor E5 Family

High throughput inference on existing server class infrastructure

Leadership Throughput

- Classify 1115 images/second

Server Class Reliability

- Industry standard server features: high reliability, hardware enhanced security

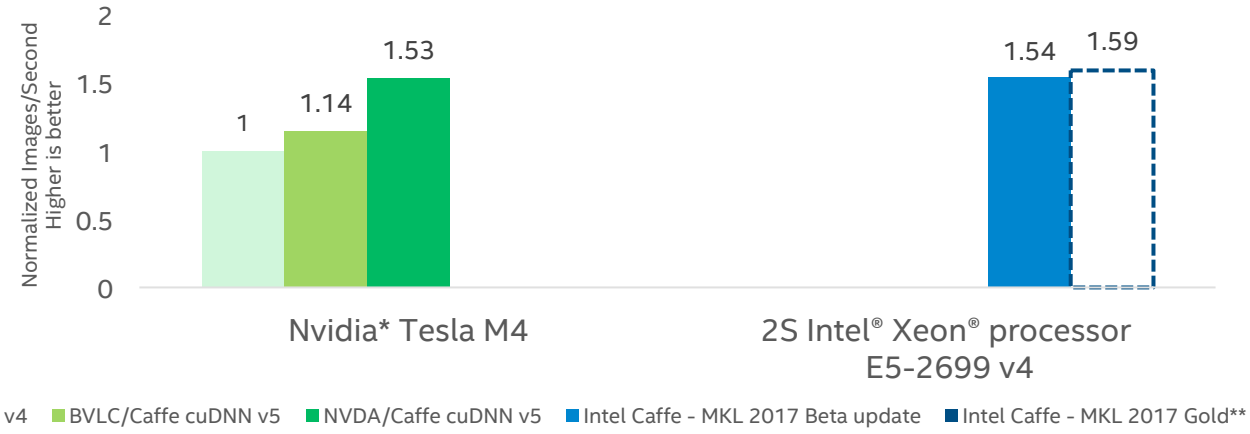
Lowest TCO With Good Infrastructure Flexibility

- Standard server infrastructure
- Open standards, libraries & frameworks
- Optimized to run wide variety of data center workloads

Configuration: 2S Intel® Xeon® Processor E5-2699 v4, 22C, 2.3GHz, 128GB, Red Hat Enterprise Linux* 6.7, Intel® Caffe : https://github.com/intelcaffe
AlexNet: https://papers.nips.cc/paper/4824-Large image database-classification-with-deep-convolutional-neural-networks.pdf, Batch Size:256; ** Q3'16 is estimated based on MKL engineering version
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/performance
Cost reduction scenarios described are intended as examples of how a given Intel- based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction. Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance. *Other names and brands may be claimed as property of others.

Scoring on Intel® Xeon® Processor E5-2699 v4

Deep Learning Image Classification SINGLE-NODE SCORING Performance



Topology: **Caffe*/AlexNet**[1] Dataset: **Large image database**

Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance . *Other names and brands may be property of others
Configurations (more details see slide 10):

1. Nvidia Tesla M4* (core@923MHz, 4GB, mem@3004MHz, 75W), DIGITS* Deep Learning Machine , 1S Intel® Xeon® Processor E5-2620 v3, 2.4GHz, 64GB, Ubuntu 14.04, Nvidia* Driver version 352.68, cuDNN v4, BVLC/ Caffe cuDNN v5 or NVIDIA/Caffe cuDNN v5
2. 2S Intel® Xeon® Processor E5-2699 v4, 22C, 2.3GHz, 128GB, Red Hat Enterprise Linux* 6.7, Intel® Caffe : https://github.com/intelcaffe

AlexNet: https://papers.nips.cc/paper/4824-Large image database-classification-with-deep-convolutional-neural-networks.pdf, Batch Size:256; ** Q4'16 based on MKL engineering version

TOOL AND LIBRARY DETAILS

INTEL® DAAL

Intel® Data Analytics Acceleration Library (Intel® DAAL)

An Intel-optimized library that provides building blocks for all data analytics stages, from data preparation to data mining & machine learning

- Python, Java & C++ APIs
- Can be used with many platforms (Hadoop*, Spark*, R*, …) but not tied to any of them
- Flexible interface to connect to different data sources (CSV, SQL, HDFS, …)
- Windows*, Linux*, and OS X*

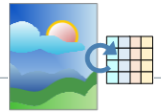
- Developed by same team as the industry-leading Intel® Math Kernel Library
- Open source, Free community and commercial premium-supported options
- Also included in Parallel Studio XE suites

Intel DAAL Overview

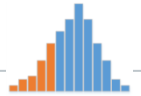
Industry leading performance, C++/Java/Python library for machine learning and deep learning optimized for Intel® Architectures.

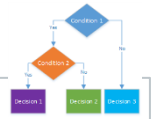
Pre-processing → **Transformation** → **Analysis** → **Modeling** → **Validation** → **Decision Making**

(De-)Compression



PCA
Statistical moments
Variance matrix
QR, SVD, Cholesky
Apriori





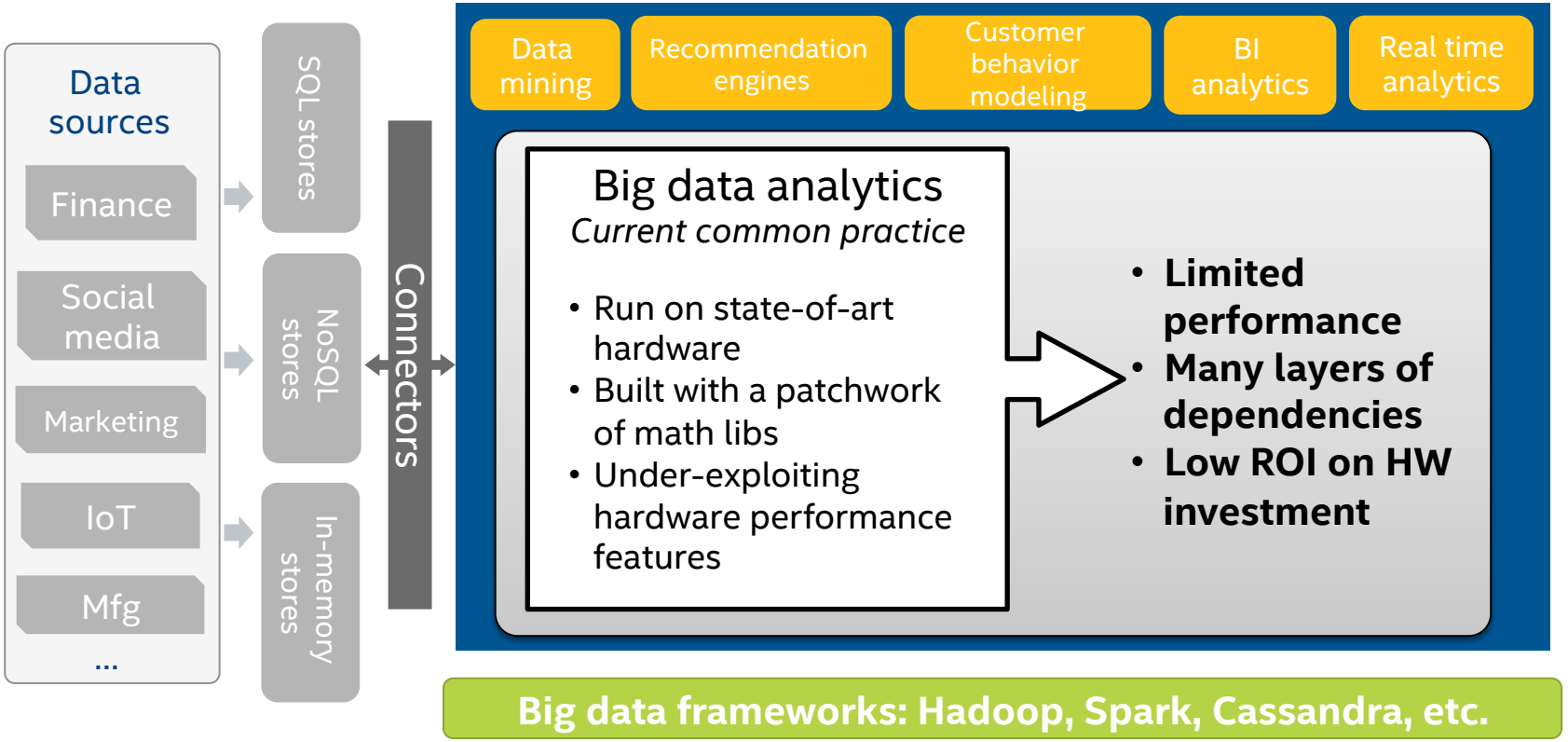
Linear regression
Naïve Bayes
SVM
Classifier boosting

Kmeans
EM GMM

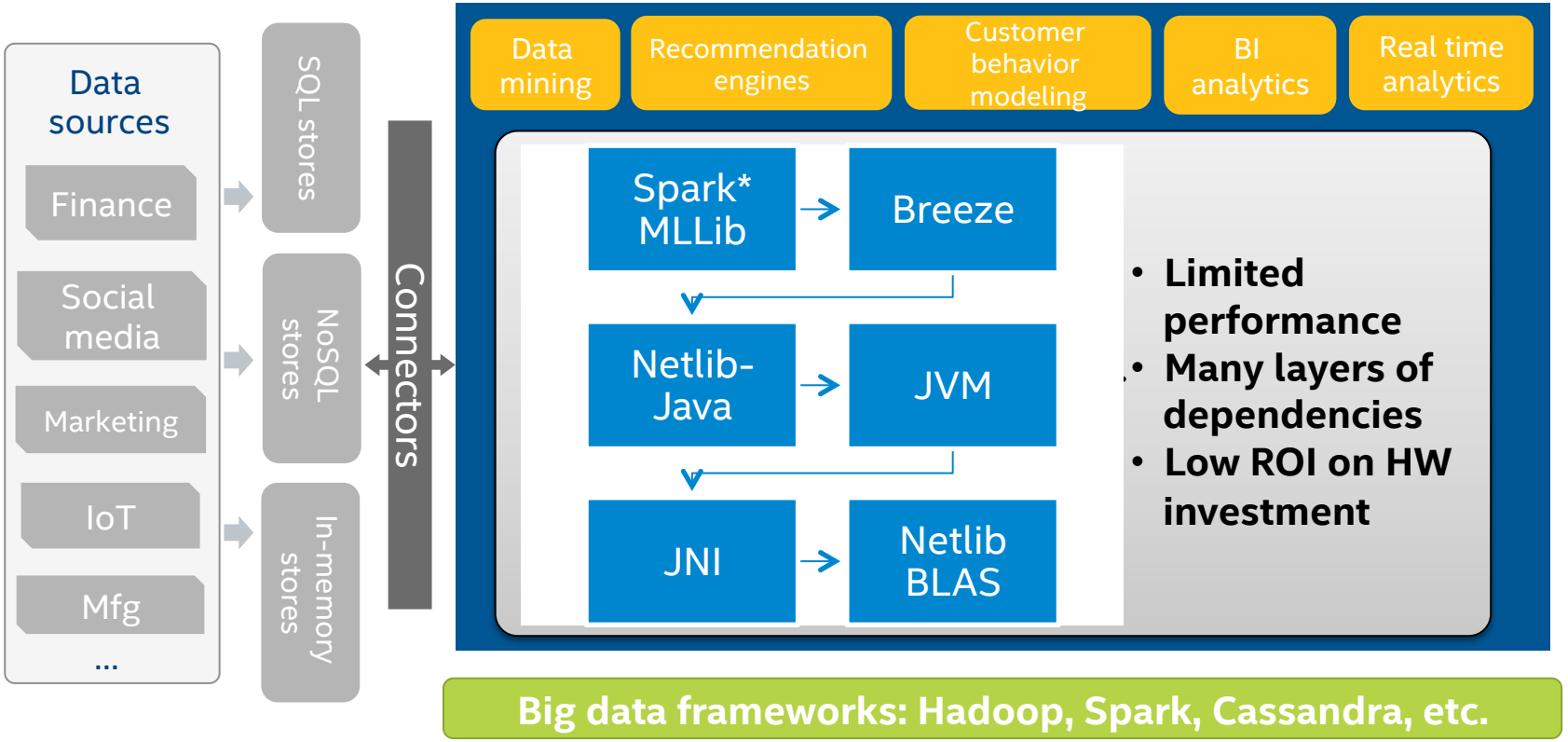
Collaborative filtering

Neural Networks

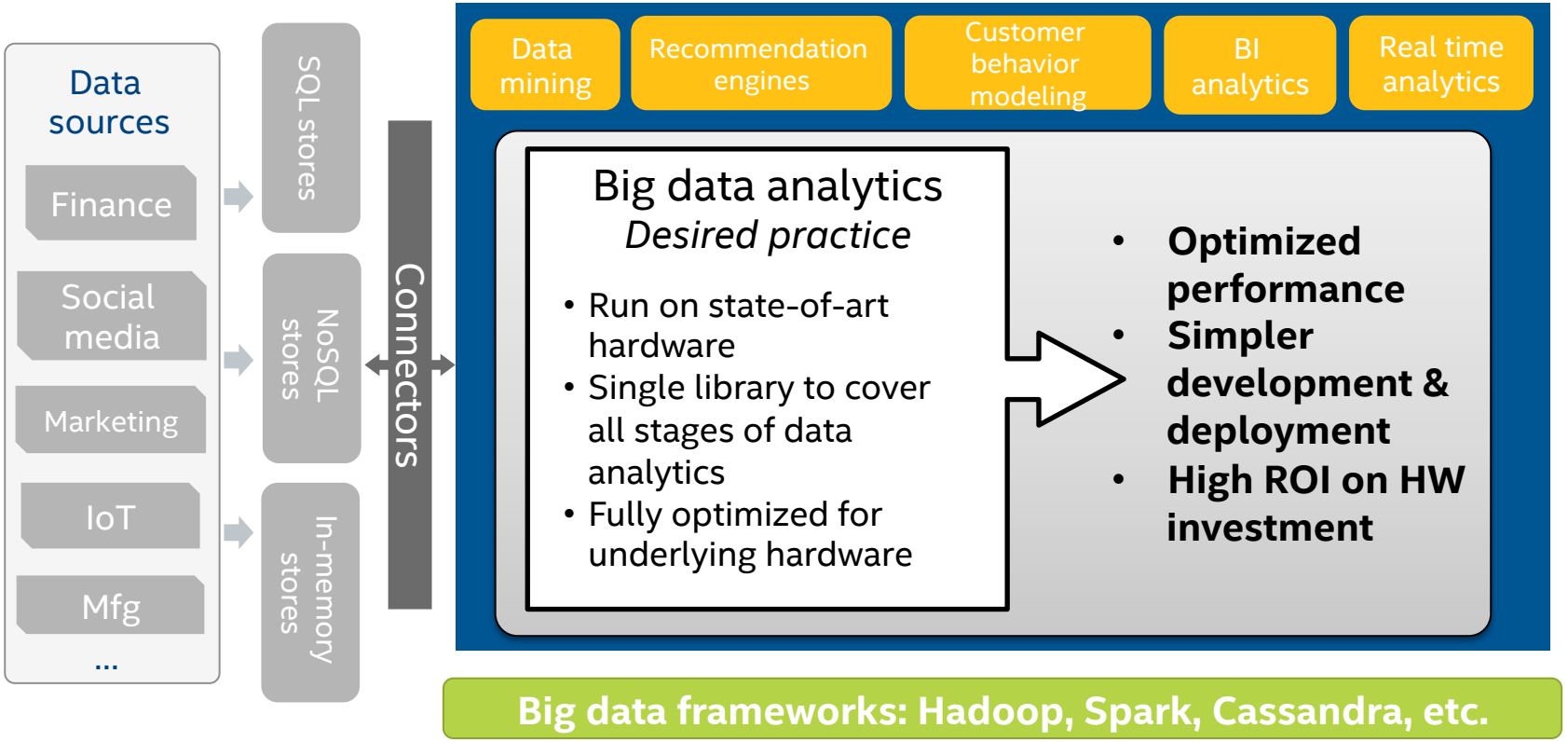
Problem Statement



Problem Statement

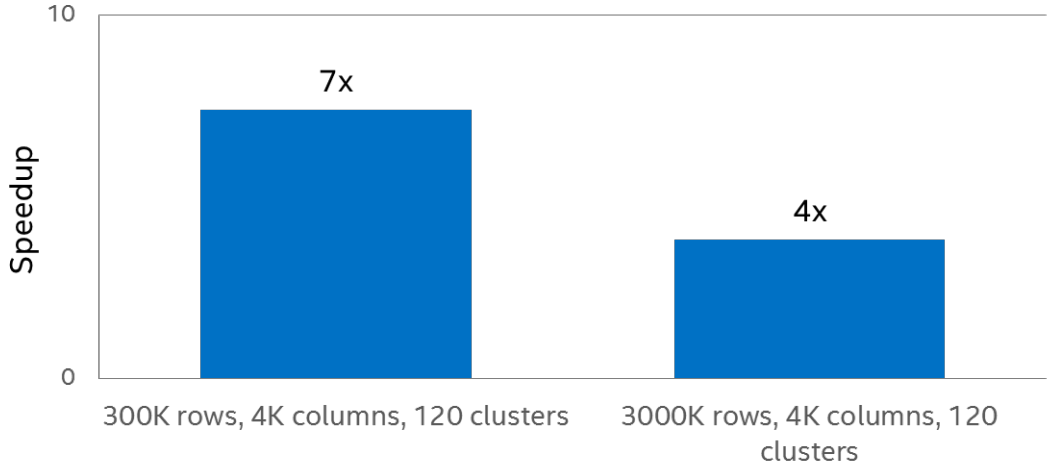


Desired Solution



Intel® DAAL vs. Spark* Mllib

K-means Performance Comparison on Eight-node Cluster



Configuration Info - Versions: Intel® Data Analytics Acceleration Library 2017, Spark 1.2; Hardware: Intel® Xeon® Processor E5-2699 v3, 2 Eighteen-core CPUs (45MB LLC, 2.3GHz), 128GB of RAM per node; Operating System: CentOS 6.6 x86_64.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. * Other brands and names are the property of their respective owners. Benchmark Source: Intel Corporation

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804 .

Intel DAAL Components

Data Management

Interfaces for data representation and access. Connectors to a variety of data sources and data formats, such HDFS, SQL, CSV, ARFF, and user-defined data source/format

Data Sources

Numeric Tables

Compression / Decompression

Serialization / Deserialization

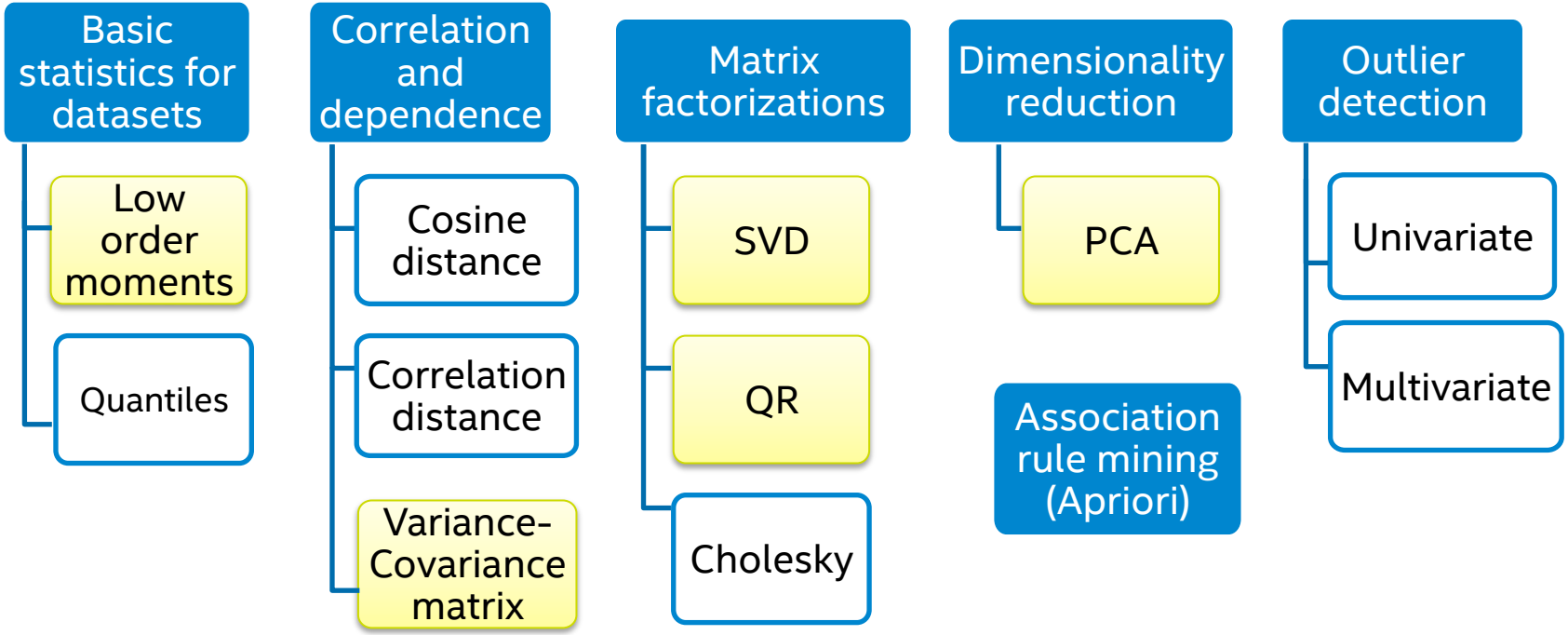
Data Processing Algorithms

Optimized analytics building blocks for all data analysis stages, from data acquisition to data mining and machine learning

Data Modeling Algorithms

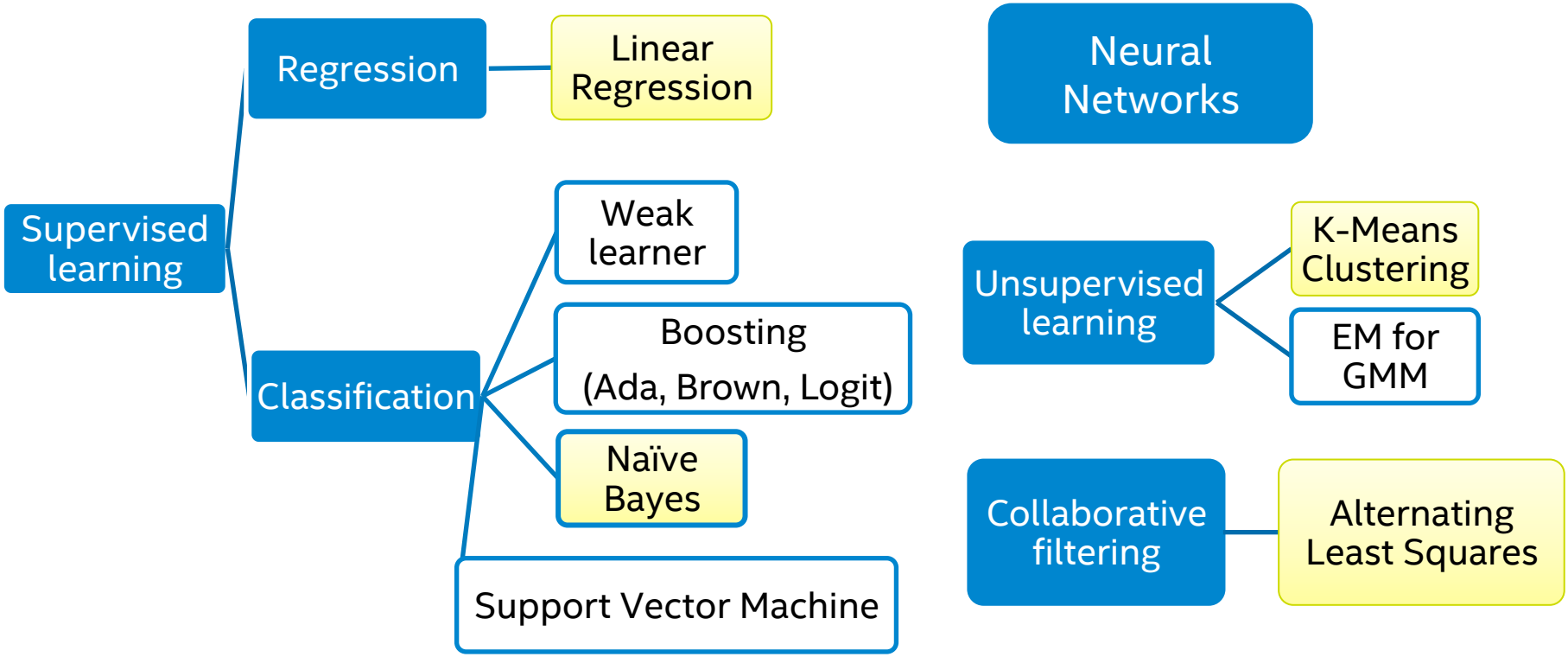
Data structures for model representation, and operations to derive model-based predictions and conclusions

Data Transformation and Analysis in Intel® DAAL



Algorithms supporting streaming and distributed processing in initial release

Machine Learning in Intel® DAAL



Algorithms supporting streaming and distributed processing

What's New: Intel DAAL 2017

- Neural Networks
- Python API (a.k.a. PyDAAL)
  - Easy installation through Anaconda or pip
- Open source project on GitHub

Fork me on GitHub:
https://github.com/01org/daal

INTEL® MKL AND MKL-DNN

Intel® Math Kernel Library (Intel® MKL) Introduction

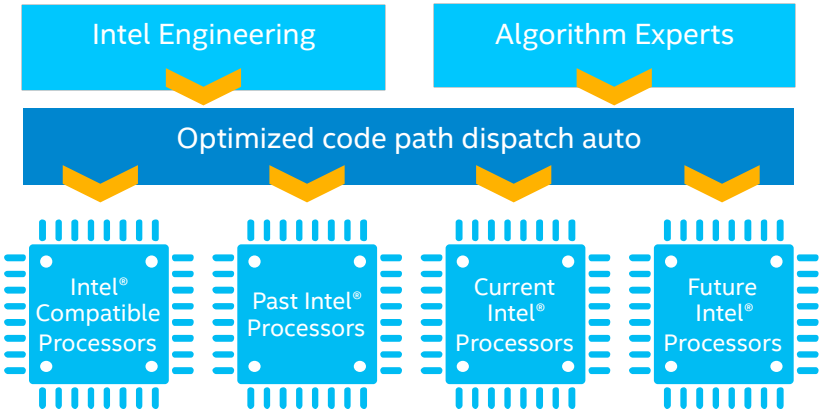
Highly optimized threaded math routines

- Performance, Performance, Performance!

Industry's leading math library

- Widely used in science, engineering, data processing

Tuned for Intel® processors – current and next generation



Evans Data Corporation EDC

EDC North America Development Survey 2016, Volume I

More math library users depend on MKL than any other library

Be multiprocessor aware

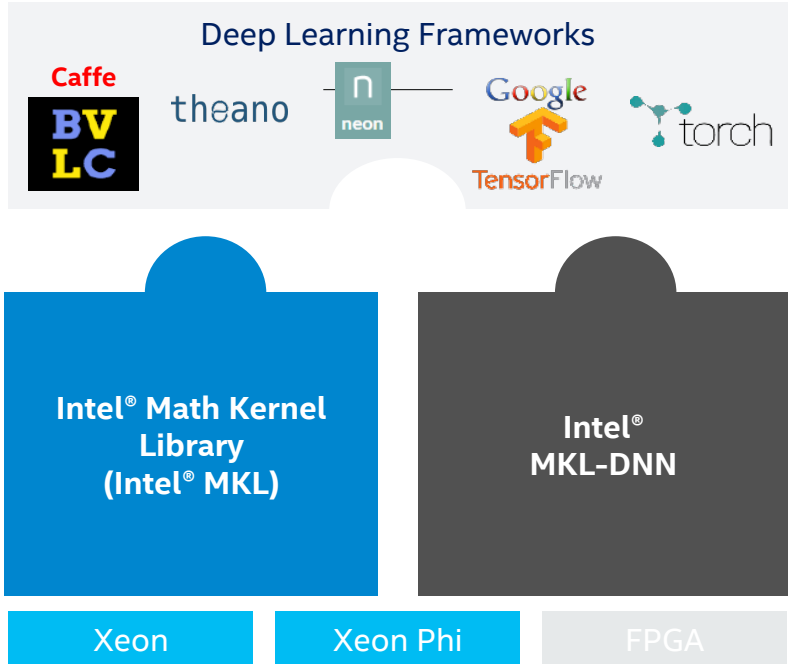
- Cross-Platform Support
- Be vectorised , threaded, and distributed multiprocessor aware

Components of Intel MKL 2017

New

Linear Algebra	Fast Fourier Transforms	Vector Math	Summary Statistics	And More…	Deep Neural Networks
• BLAS • LAPACK • ScaLAPACK • Sparse BLAS • Sparse Solvers • Iterative • PARDISO* • Cluster Sparse Solver	• Multidimensional • FFTW interfaces • Cluster FFT	• Trigonometric • Hyperbolic • Exponential • Log • Power • Root • Vector RNGs	• Kurtosis • Variation coefficient • Order statistics • Min/max • Variance-covariance	• Splines • Interpolation • Trust Region • Fast Poisson Solver	• Convolution • Pooling • Normalization • ReLU • Inner Product

Intel® Math Kernel Library and Intel® MKL-DNN for Deep Learning Framework Optimization

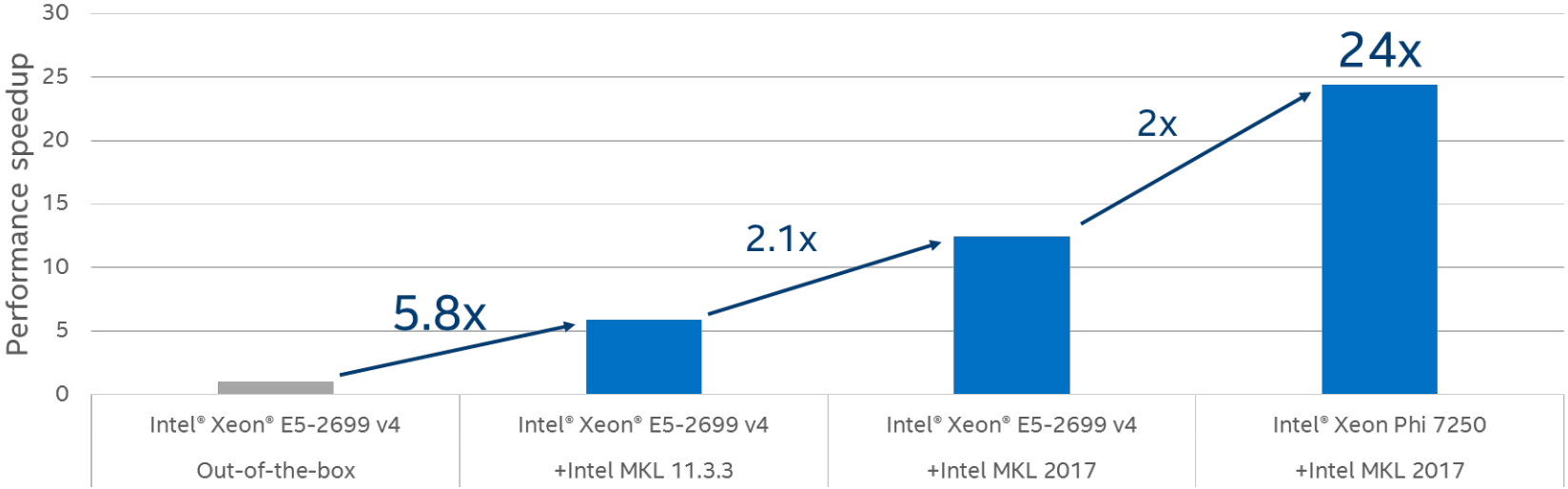


Intel® MKL	Intel® MKL-DNN
DNN primitives + wide variety of other math functions	DNN primitives
C DNN APIs	C/C++ DNN APIs
Binary distribution	Open source DNN code*
Free community license. Premium support available as part of Parallel Studio XE	Apache 2.0 license
Broad usage DNN primitives; not specific to individual frameworks	Multiple variants of DNN primitives as required for framework integrations
Quarterly update releases	Rapid development ahead of Intel MKL releases

* GEMM matrix multiply building blocks are binary

Improved Deep Neural Network training performance using Intel® Math Kernel Library (Intel® MKL)

Caffe/AlexNet single node training performance

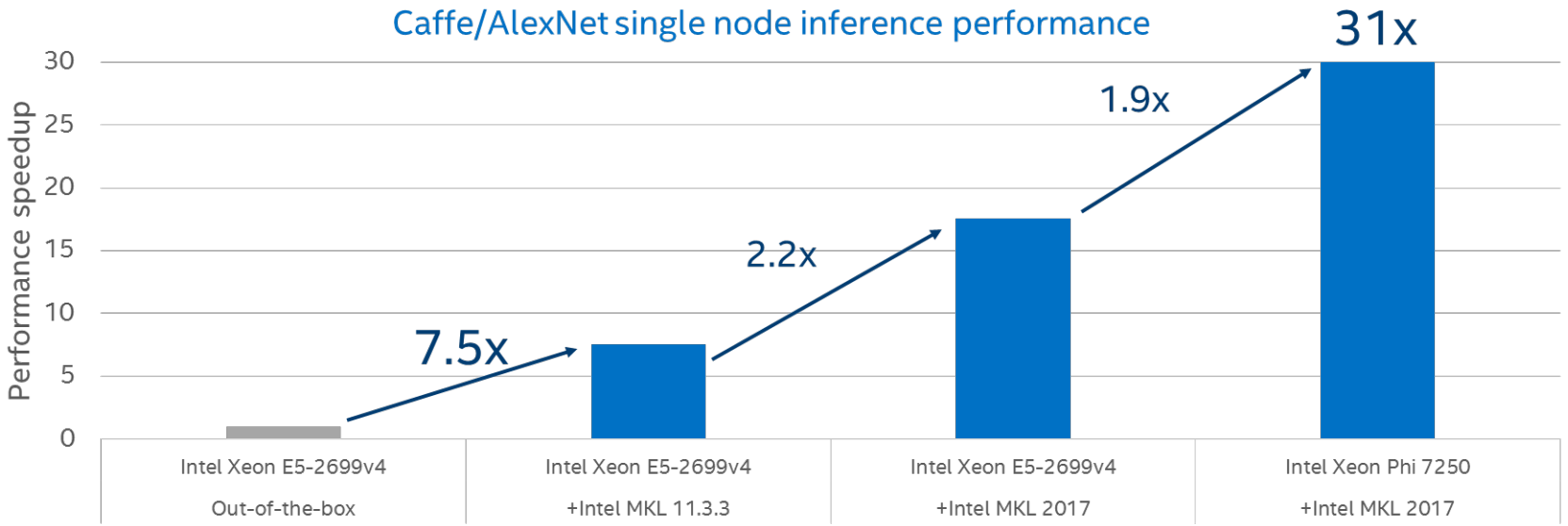


Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance. *Other names and brands may be property of others
Configurations:

- 2 socket system with Intel® Xeon Processor E5-2699 v4 (22 Cores, 2.2 GHz,), 128 GB memory, Red Hat* Enterprise Linux 6.7, BVLC Caffe, Intel Optimized Caffe framework, Intel® MKL 11.3.3, Intel® MKL 2017
- Intel® Xeon Phi™ Processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM), 128 GB memory, Red Hat* Enterprise Linux 6.7, Intel® Optimized Caffe framework, Intel® MKL 2017

All numbers measured without taking data manipulation into account.
Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804 .

Improved Deep Neural Network inference performance using Intel® Math Kernel Library (Intel® MKL)



Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more information go to http://www.intel.com/performance . *Other names and brands may be property of others

Configurations:

- 2 socket system with Intel® Xeon® Processor E5-2699 v4 (22 Cores, 2.2 GHz,), 128 GB memory, Red Hat* Enterprise Linux 6.7, BVLC Caffe, Intel Optimized Caffe framework, Intel® MKL 11.3.3, Intel® MKL 2017
- Intel® Xeon Phi™ Processor 7250 (68 Cores, 1.4 GHz, 16GB MCDRAM), 128 GB memory, Red Hat* Enterprise Linux 6.7, Intel® Optimized Caffe framework, Intel® MKL 2017

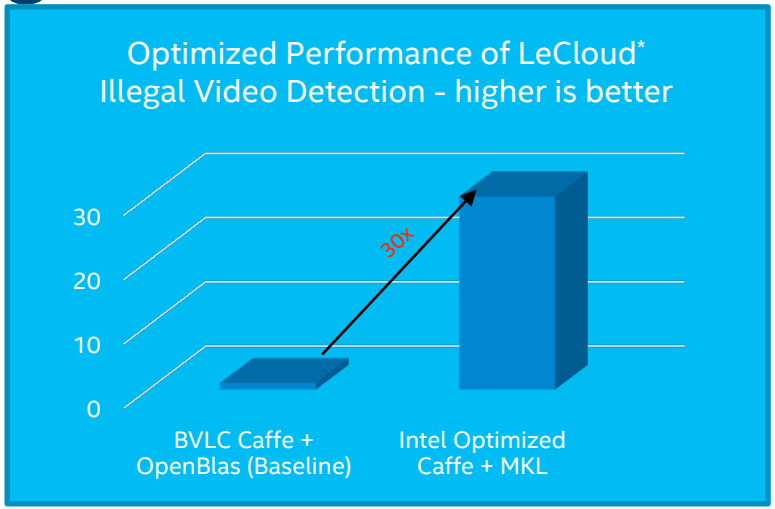
All numbers measured without taking data manipulation into account.

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804 .

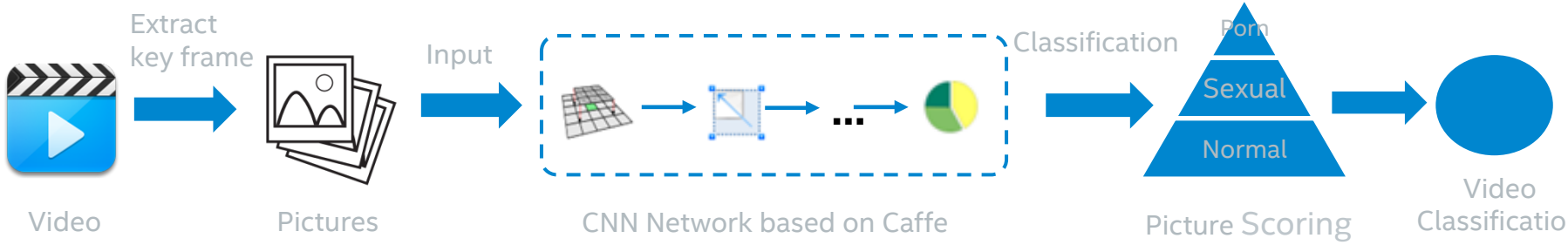
Case Study I: Deep Learning

LeCloud Illegal Video Detection*

- LeCloud : leading video cloud provider in China who provides illegal video detection service
- Originally: Adopted open source BVLC Caffe w/OpenBlas as CNN framework
- Now: Using Intel Optimized Caffe plus Intel® Math Kernel Library, achieved 30x performance improvement for training in production



* The test data is based on Intel® Xeon® E5 2680 V3 processor



Intel® DAAL+ Intel® MKL = Complementary Big Data Libraries Solution

Intel MKL	Intel DAAL
C and Fortran API Primitive level	Python, Java & C++ API High-level
Processing of homogeneous data in single or double precision	Processing heterogeneous data (mix of integers and floating point), internal conversions are hidden in the library
Type of intermediate computations is defined by type of input data (in some library domains higher precision can be used)	Type of intermediate computations can be configured independently of the type of input data
Most of MKL supports batch computation mode only	3 computation modes: Batch, streaming and distributed
Cluster functionality uses MPI internally	Developer chooses communication method for distributed computation (e.g. Spark, MPI, etc.) Code samples provided.

"Initially, the Spark/Shark-based solution required 40 hours to compete a computation. Youku improved performance significantly by implementing Intel® Math Kernel Library (Intel® MKL) into its solution...After implementation of Intel MKL, Youku reduced the computation time to less than three hours."

Source: Youku Tudou Video Sharing Recommendation Case Study

INTEL® DEEP LEARNING SDK

Intel® Deep Learning SDK

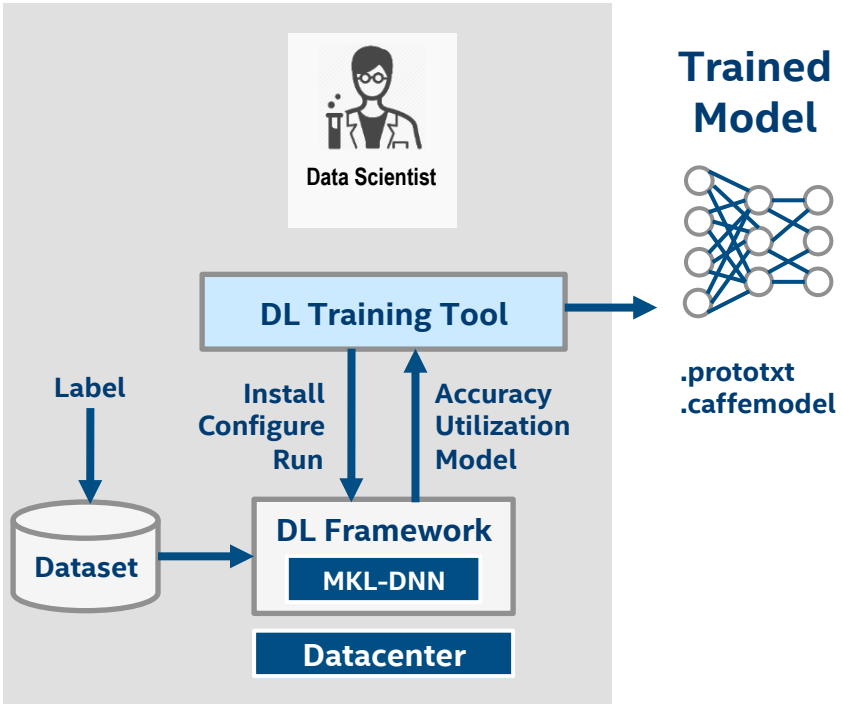
Accelerate Your Deep Learning Solution

A free set of tools for data scientists and software developers to develop, train, and deploy deep learning solutions

"Plug & Train/Deploy"	Maximum Performance	Increased Productivity
Simplify installation & preparation of deep learning models using popular deep learning frameworks on Intel hardware	Optimized performance for training and inference on Intel® Architecture	Faster Time-to-market for training and inference, Improve model accuracy, Reduce total cost of ownership

Deep Learning Training Tool

Intel® Deep Learning SDK



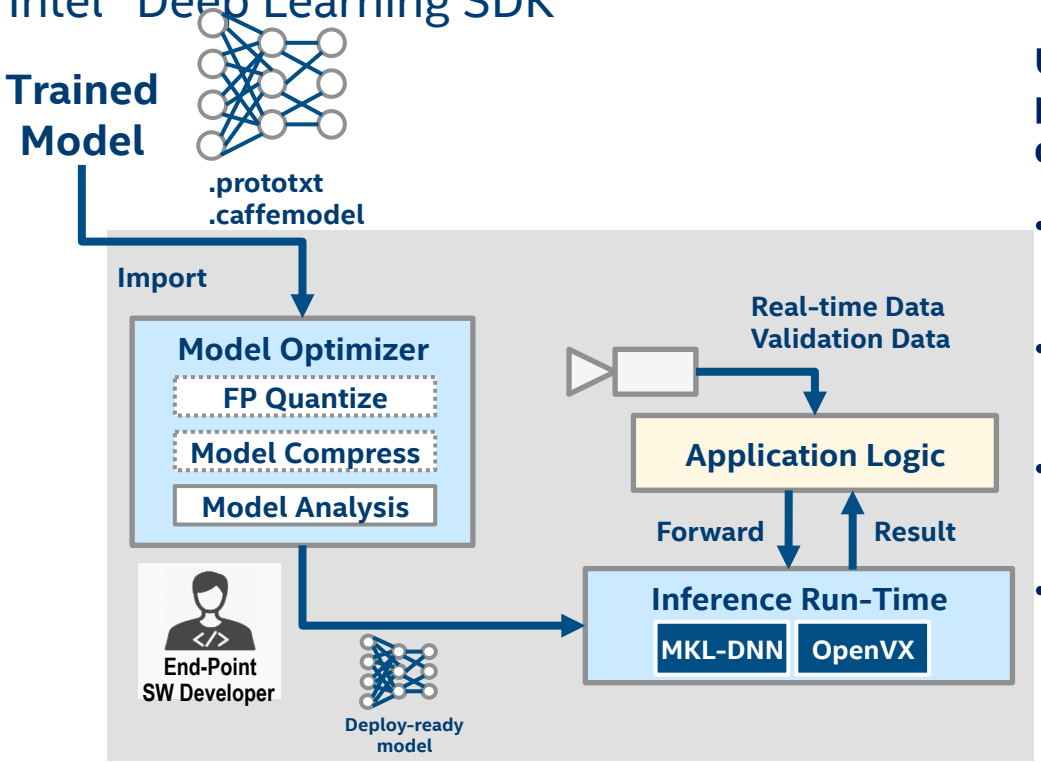
Trained Model

.prototxt
.caffemodel

- Simplify installation of Intel optimized Deep Learning Frameworks
- Easy and Visual way to Set-up, Tune and Run Deep Learning Algorithms:
  - ✓ Create training dataset
  - ✓ Design model with automatically optimized hyper-parameters
  - ✓ Launch and monitor training of multiple candidate models
  - ✓ Visualize training performance and accuracy

Deep Learning Deployment Tool

Intel® Deep Learning SDK



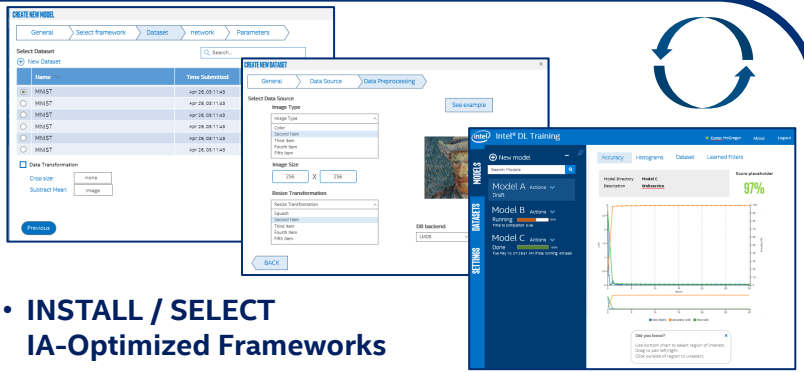
Unleash fast scoring performance on Intel products while abstracting the HW from developers

- Imports trained models from all popular DL framework regardless of training HW
- Compresses model for improved execution, storage & transmission (pruning, quantization)
- Generates scoring HW-specific code (C/C++, OpenVX graphs, OpenCL, etc.)
- Enables seamless integration with full system / application software stack

Deep Learning Tools for End-to-End Workflow

Intel® Deep Learning SDK

Intel DL Training Tool



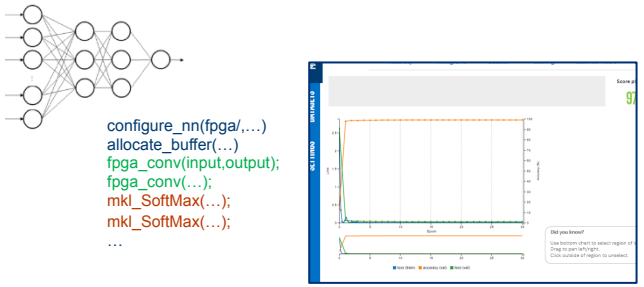
- **INSTALL / SELECT IA-Optimized Frameworks**
- **PREPARE / CREATE Dataset with Ground-truth**
- **DESIGN / TRAIN Model(s) with IA-Opt. Hyper-Parameters**
- **MONITOR Training Progress across Candidate Models**
- **EVALUATE Results and ITERATE**

MKL-DNN Optimized Machine Learning Frameworks

theano Caffe Microsoft Google TensorFlow torch

Xeon (local or cloud)

Intel DL Deployment Tool

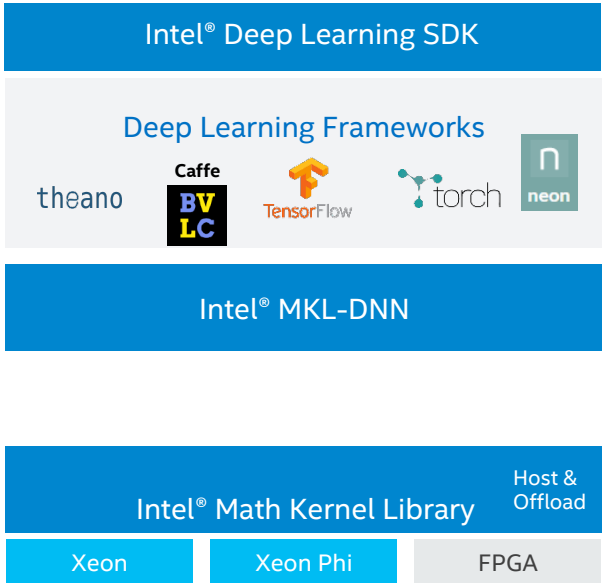


- **IMPORT Trained Model (trained on Intel or 3rd Party HW)**
- **COMPRESS Model for Inference on Target Intel HW**
- **GENERATE Inference HW-Specific Code (OpenVX, C/C++)**
- **INTEGRATE with System SW / Application Stack & TUNE**
- **EVALUATE Results and ITERATE**

Optimized libraries & run-times (MKL-DNN, OpenVX, OpenCL)
Data acquisition (sensors) and acceleration HW (FPGA, etc)

Target Inference Hardware Platform (physical or simulated)

Intel Deep Learning Software Stack



Tools to accelerate design, training and deployment of deep learning solutions

Targeted release: early Q4'2016

Popular Deep Learning frameworks

Open source Intel x86 optimized DNN APIs, combined with Intel® MKL and build tools designed for scalable, high-velocity integration with ML/DL frameworks.

Includes:

- New algorithms ahead of MKL releases
- IA optimizations contributed by community

SW building block to extract max Intel HW performance and provide common interface to all Intel accelerators.

Intel libraries as path to bring optimized ML/DL frameworks to Intel hardware

*Other names and brands may be claimed as property of others.

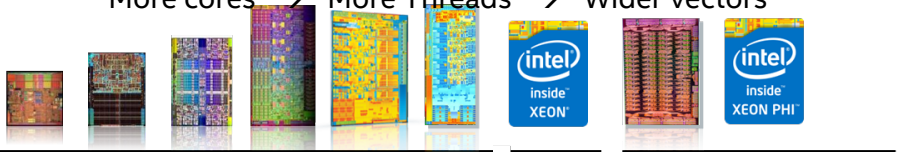
Software.intel.com/machine-learning

INTEL® DISTRIBUTION FOR PYTHON

OUR APPROACH

1. Enable hooks to Intel® MKL, Intel® DAAL, Intel® IPP functions in the most popular numerical packages
   - NumPy, SciPy, Scikit-Learn, PyTables, Scikit-Image, …
2. Available through Intel® Distribution for Python* and as Conda packages
   - Most optimizations eventually upstreamed to home open source projects
3. Provide Python interfaces for Intel® DAAL (a.k.a PyDAAL)

More cores → More Threads → Wider vectors



	Intel® Xeon® Processor 64-bit	Intel® Xeon® Processor 5100 series	Intel® Xeon® Processor 5500 series	Intel® Xeon® Processor 5600 series	Intel® Xeon® Processor E5-2600 v2 series	Intel® Xeon® Processor E5-2600 v3 series	~	Future Intel® Xeon® Processor[1]	Intel® Xeon Phi™ x100 Coprocessor	Intel® Xeon Phi™ x200 Processor & Coprocessor
Up to Core(s)	1	2	4	6	12	18	~	*Tbd*	57-61	TBD
Up to Threads	2	2	8	12	24	36		*tbd*	228-244	TBD
SIMD Width	128	128	128	128	256	256	~	512	512	512
Vector ISA	Intel® SSE3	Intel® SSE3	Intel® SSE4.2	Intel® AVX	Intel® AVX	Intel® AVX2		Intel® AVX-512	IMCI 512	Intel® AVX-512

Numpy & Scipy optimizations with Intel® MKL

Configuration info: - Versions: Intel® Distribution for Python 2017 Beta, icc 15.0; Hardware: Intel® Xeon® CPU E5-2698 v3 @ 2.30GHz (2 sockets, 16 cores each, HT=OFF), 64 GB of RAM, 8 DIMMS of 8GB@2133MHz; Operating System: Ubuntu 14.04 LTS.

Linear Algebra

- BLAS
- LAPACK
- ScaLAPACK
- Sparse BLAS
- Sparse Solvers
  - Iterative
  - PARDISO SMP & Cluster

Up to 100x faster

Fast Fourier Transforms

- Multidimensional
- FFTW interfaces
- Cluster FFT

Up to 10x faster!

Vector Math

- Trigonometric
- Hyperbolic
- Exponential
- Log
- Power
- Root

Up to 10x faster!

Vector RNGs

- Multiple BRNG
- Support methods for independent streams creation
- Support all key probability distributions

Up to 60x faster!

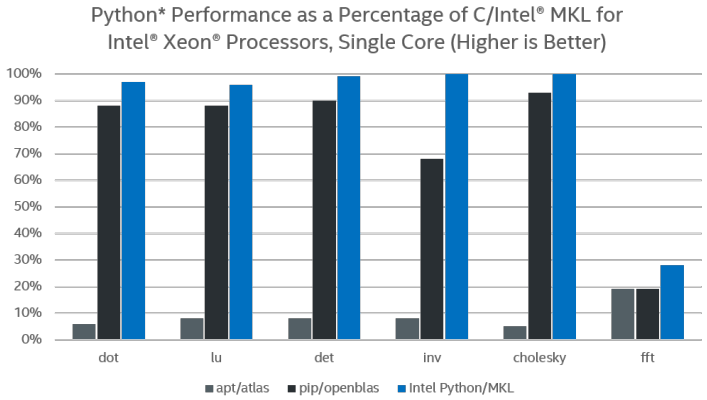
Summary Statistics

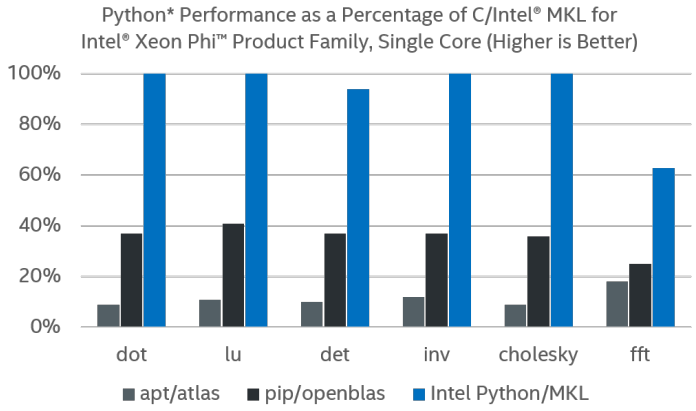
- Kurtosis
- Variation coefficient
- Order statistics
- Min/max
- Variance-covariance

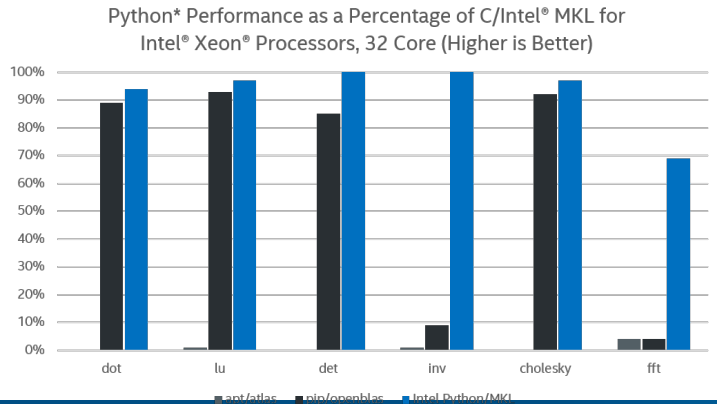
And More

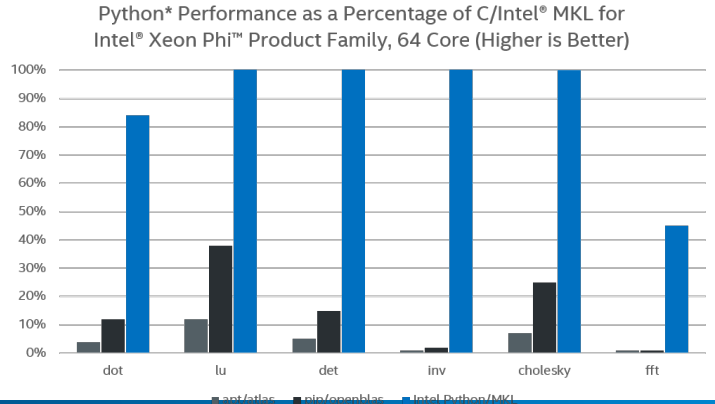
- Splines
- Interpolation
- Trust Region
- Fast Poisson Solver

Near native performance on Intel® Xeon™ and Intel® Xeon Phi™









Hardware/Problem Size	dot	lu	det	inv	cholesky	fft
Intel® Xeon® processor (32 core) and Intel® Xeon Phi™ processor (64 core)	(20k, 10k) and (10k, 20k)	(35k, 35k)	(15k, 15k)	(25k, 25k)	(40k, 40k)	520k
Intel Xeon processor (1 core)	(20k, 5k) and (5,20k)	(20k, 20k)		(10k, 10k)	(10k, 10k)	
Intel Xeon Phi processor (1 core)	(20k, 300) and (300, 20k)	(6k, 6k)	(4k, 4k)	(2k, 2k)		

Configuration Info: apt/atlas: installed with apt-get, Ubuntu* 16.10, Python* 3.5.2, numpy* 1.11.0, scipy* 0.17.0; pip*/openblas*: installed with pip, Ubuntu 16.10, python 3.5.2, numpy 1.11.1, scipy 0.18.0; Intel Python: Intel® Distribution for Python 2017;, Hardware: Intel Xeon processor: Intel Xeon processor E5-2698 v3 @ 2.30 GHz (2 sockets, 16 cores each, HT=off), 64 GB of RAM, 8 DIMMS of 8GB@2133MHz; Intel Xeon Phi processor: Intel Intel® Xeon Phi™ processor 7210 1.30 GHz, 96 GB of RAM, 6 DIMMS of 16GB@1200MHz

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. * Other brands and names are the property of their respective owners. Benchmark source: Intel Corporation.

Optimization Notice: Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804

- Runs out-of-the-box with any Python
- Intel Distribution for Python delivers much greater efficiency than “system” Python
- Potential for future multi-threaded performance tunings in numpy and scipy

Roadmap & Reviews

"I expected Intel's numpy to be fast but it is significant that plain old python code is much faster with the Intel version too."

Dr. Donald Kinghorn, Puget Systems Review

InfoWorld

Intel's Python distribution provides a major math boost

The still-in-beta Python distribution uses Math Kernel Library to speed up processing on Intel hardware
The distribution's main touted advantage is speed -- but not a PyPy-style general speedup via a JIT. Instead, the MKL speeds up certain math operations so that they run faster on one thread and multiple threads.

Available as free standalone download

Commercial support through Intel® Parallel Studio 2017

insideHPC

HPC Podcast Looks at Intel's Pending Distribution of Python

Yes, Intel is doing their own Python build! It is still in beta but I think it's a great idea. .....Yeah, it's important!

Download at https://software.intel.com/en-us/python-distribution

CALL-TO-ACTION

Summary

I. Deep Learning framework optimizations on Xeon, Xeon Phi – Session #

II. Intel DAAL

III. Intel MKL, MKL-DNN

IV. Intel Python optimizations – Session #

V. Intel Deep Learning SDK

Learn more at www.intel.com/machinelearning

Legal Disclaimer & Optimization Notice

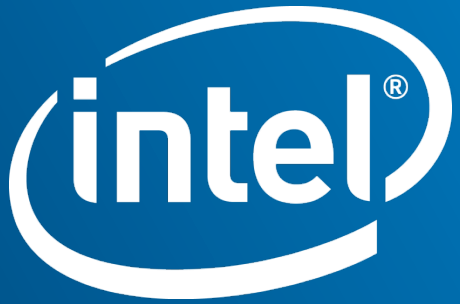
INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS”. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Optimization Notice

Intel’s compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804


intel®