BLOG   03.07.19

INTEL OPTIMIZED DATA SCIENCE VIRTUAL MACHINE ON MICROSOFT AZURE*

In response to the surge in popularity of AI and machine learning, Cloud Service Providers (CSPs) have begun providing virtual machines (VMs) specialized for these applications. However, the default offerings usually contain unoptimized machine learning frameworks that do not leverage the [Intel® Advanced Vector Extensions 512 (Intel® AVX-512)](#) instruction set for faster vector operations on Intel® Xeon® Scalable processors. To address this, Intel has [collaborated with Microsoft](#) to build the Intel Optimized Data Science Virtual Machine (DSVM), an extension of the Ubuntu* version of [Azure* DSVMs](#) with CPU-optimized conda environments for TensorFlow* and MXNet*. These optimized environments require no modification to existing TensorFlow or MXNet code, and provide an average of 7.7X speedup over unoptimized environments (see Figure 1).

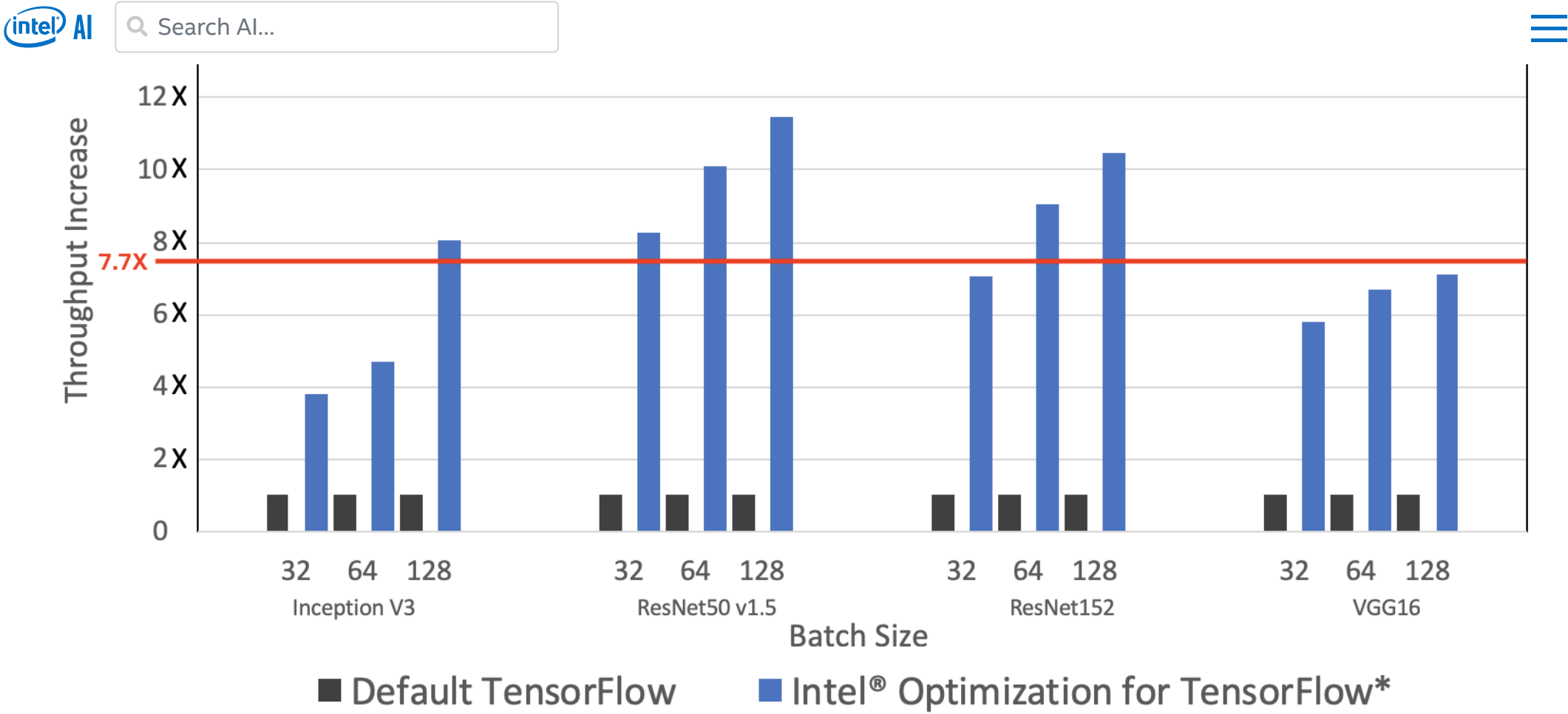## Default TensorFlow vs. Intel® Optimization for TensorFlow*

Figure 1: Intel® Optimization for TensorFlow provides an average of 7.7X increase (average indicated by the red line) in training throughput on major CNN topologies. Performance results are based on testing as of 01/15/2019 by Intel. Please see complete testing configuration in disclaimers. To run your own benchmarks using tf_cnn_benchmarks, see https://github.com/IntelAI/azure-applications/tree/master/scripts/benchmark.

In this article, we'll outline three easy steps to launch an Intel Optimized DSVM of your own. Before getting started, make sure you've created an Azure account and can access https://portal.azure.com/. To create an account, see https://azure.microsoft.com.

STEP 1

Choose the **Intel Optimized Data Science VM for Linux**

1. Once your account is created, navigate to https://portal.azure.com. In the top left, select "**Create a resource**", search "Intel Optimized Data Science VM for Linux", then click "**Create**" at the bottom of the screen:

STEP 2

Follow prompts to configure the DSVM parameters. Once you've clicked **Create**, you'll be prompted to set some parameters:

1. **Virtual machine name:** Name the VM.

2. **Username and password:** Passwords simplify connecting to the VM.

3. **Subscription:** Make sure the appropriate subscription is selected.

4. **Resource group:** Select an existing resource group.

5. **Location:** The default region should work just fine or choose your desired location.

Create Intel® Optimized Data... ✕

Basics ☐ ✕

(intel) AI   🔍 Search AI...

1   Basics
Configure basic settings   ❯

2   Settings
Configure additional settings   ❯

3   Summary
Intel® Optimized Data Science...   ❯

4   Buy   ❯

---

* **Name** ℹ

| name | ✓ |

User name ℹ

| azuser | ✓ |

Password ℹ

| •••••••••••• | ✓ |

Confirm password

| •••••••••••• | ✓ |

Subscription

| Deep Learning Azure | ⌄ |

* **Resource group** ℹ

| mattson-skl-rg | ⌄ |

Create new

* **Location**

| East US | ⌄ |

When completely filled out, clicking **OK** will advance to the next screen, where you can select the VM size:

Create Intel® Optimized Data... ✕

Settings ✕

1   Basics
Done   ✓

2   Settings
Configure additional settings   ❯

3   Summary
Intel® Optimized Data Science...   ❯

4   Buy   ❯

* Virtual machine size ℹ
1x Standard F8s_v2   ❯

Choose a size

Browse the available sizes and their features

Search   Compute type

| F | | Current generation ⌄ |

RECOMM... ↑↓	SKU ↑↓	TYPE ↑↓	COMPUT... ↑↓	VCPUS ↑↓	GB RAM ↑↓	DATA
Available						
	F2s_v2	Standard	Compute opti	2	4	4
	F4s_v2	Standard	Compute opti	4	8	8
★	F8s_v2	Standard	Compute opti	8	16	16
★	F16s_v2	Standard	Compute opti	16	32	32
★	F32s_v2	Standard	Compute opti	32	64	32
	F64s_v2	Standard	Compute opti	64	128	32
	F72s_v2	Standard	Compute opti	72	144	32

Clicking the VM size allows you to change the hardware on which your VM will run. We recommend the following instances powered by Intel Xeon Scalable processors to take advantage of Intel AVX-512:

1. **Compute Optimized:** Fsv2-series (F4sv2, F8sv2, F16sv2, F32sv2, F64sv2, F72sv2)

2. **High Performance Compute:** HC-series

Once you've selected a size, clicking **OK** will begin the Validation step, and you'll be able to see a summary of your VM:

Clicking **OK** again brings you to the final screen. Click **Create** to launch the instance.

LOG IN AND RUN BENCHMARKS:

Once the VM is launched, a custom extension triggers the one-time installation of Intel optimized deep learning frameworks (installation takes ~10 minutes). After launch, log in to the machine as usual and run `conda env list`. The bolded environments are optimized for execution on Intel® Xeon® processors:

```
azuser@vmname:~$ conda env list
# conda environments:
#
base                      /data/anaconda
intel_mxnet_p27           /data/anaconda/envs/intel_mxnet_p27
intel_mxnet_p36           /data/anaconda/envs/intel_mxnet_p36
intel_tensorflow_p27      /data/anaconda/envs/intel_tensorflow_p27
intel_tensorflow_p36      /data/anaconda/envs/intel_tensorflow_p36
py35                    * /data/anaconda/envs/py35
py36                      /data/anaconda/envs/py36
```

To verify the presence of [Intel® Math Kernel Library for Deep Neural Networks (Intel® MKL-DNN)](#), activate an optimized environment and run the following:

```
azuser@vmname:~$ source activate intel_tensorflow_p36
```

```
| cut -d ' ' -f2;/tensorflow/libtensorflow_framework.so | grep libmklml;

MKL shared libs:        libmklml_intel.so => /data/anaconda/envs/intel_tensorflow_p36/lib/python3.6/site-
packages/tensorflow/../_solib_k8/_U@mkl_Ulinux_S_S_Cmkl_Ulibs_Ulinux___Uexternal_Smkl_Ulinux_Slib/libmklml_i
  (0x00007f27e241f000)

(intel_tensorflow_p36) azuser@vmname:~$ python -c "import tensorflow as tf;
print(tf.pywrap_tensorflow.IsMklEnabled())"

True
```

If you would like to run TensorFlow* CNN training benchmarks, [download the benchmark script](#) and run as indicated (**Note:** do not activate an optimized virtual environment before running the benchmark, this is taken care of in the script).

```
azuser@vmname:~$ wget https://raw.githubusercontent.com/IntelAI/azure-
applications/master/scripts/benchmark/intel_tf_cnn_benchmarks.sh

azuser@vmname:~$ bash intel_tf_cnn_benchmarks.sh
...
...

...
...
######### Executive Summary #########


Environment |  Network   | Batch Size | Images/Second
----------------------------------------------------------
Default     | inception3 |     128     | 6.44
Optimized   | inception3 |     128     | 52.00


#########################################
Average Intel Optimized speedup = 8X
#########################################
```

Training throughput in the Default and Optimized environments is listed per network and batch size. By default, the script runs Inception* V3 at a batch size of 128. If you would like to run the entire suite of CNN networks (Inception V3, ResNet* 50, ResNet 152, VGG*-16) and batch sizes (32, 64, 128), pass "all" as the first argument to the benchmarking script. Note that this option will take some time.

```
azuser@vmname:~$ bash intel_tf_cnn_benchmarks.sh all
```

SUMMARY

deep learning on Intel® Architecture. You can also learn more about this new offering on the [Microsoft Azure blog](#).

NOTICES AND DISCLAIMERS

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 www.intel.com/benchmarks.

1. Testing Configuration:

```
Azure Instance Size: F72s_v2
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 72
On-line CPU(s) list: 0-71
Thread(s) per core: 2
Core(s) per socket: 18
Socket(s): 2
NUMA node(s): 2
Vendor ID: GenuineIntel
CPU family: 6
Model: 85
Model name: Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz
Stepping: 4
CPU MHz: 2693.855
BogoMIPS: 5387.73
Virtualization: VT-x
Hypervisor vendor: Microsoft
Virtualization type: full
L1d cache: 32K
L1i cache: 32K
L2 cache: 1024K
L3 cache: 33792K
NUMA node0 CPU(s): 0-35
NUMA node1 CPU(s): 36-71
```

Performance results are based on testing as of 01/15/2019 by Intel and may not reflect all publicly available security updates. No product or component can be absolutely secure.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.

ptimization 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.

 intel AI

🔍 Search AI...   ☰

*Intel, the Intel logo, and Xeon are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries. *Other names and brands may be claimed as the property of others.*

AUTHORS

Mattson Thieme

Machine Learning Engineer, Artificial Intelligence Products Group

Ravi Panchumarthy

Machine Learning Engineer, Artificial Intelligence Products Group

Prashant Shah

Director of Engineering of Health and Life Sciences, Artificial Intelligence Products Group
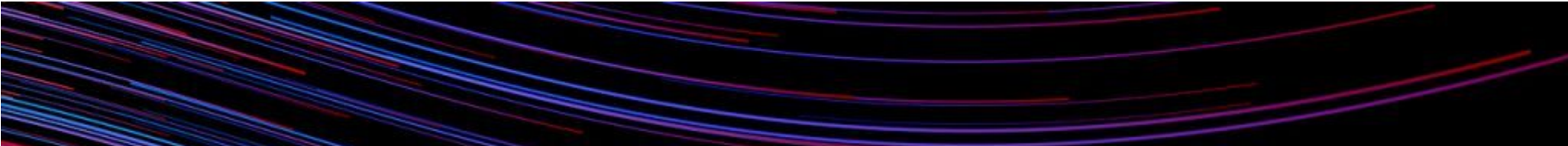
TAGS

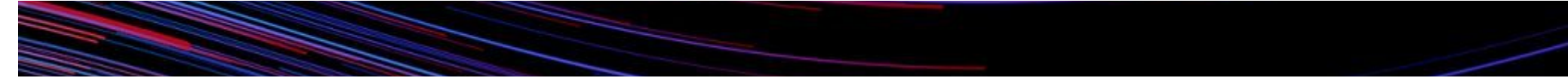General, ResNet 50, Azure, Microsoft

RELATED CONTENT
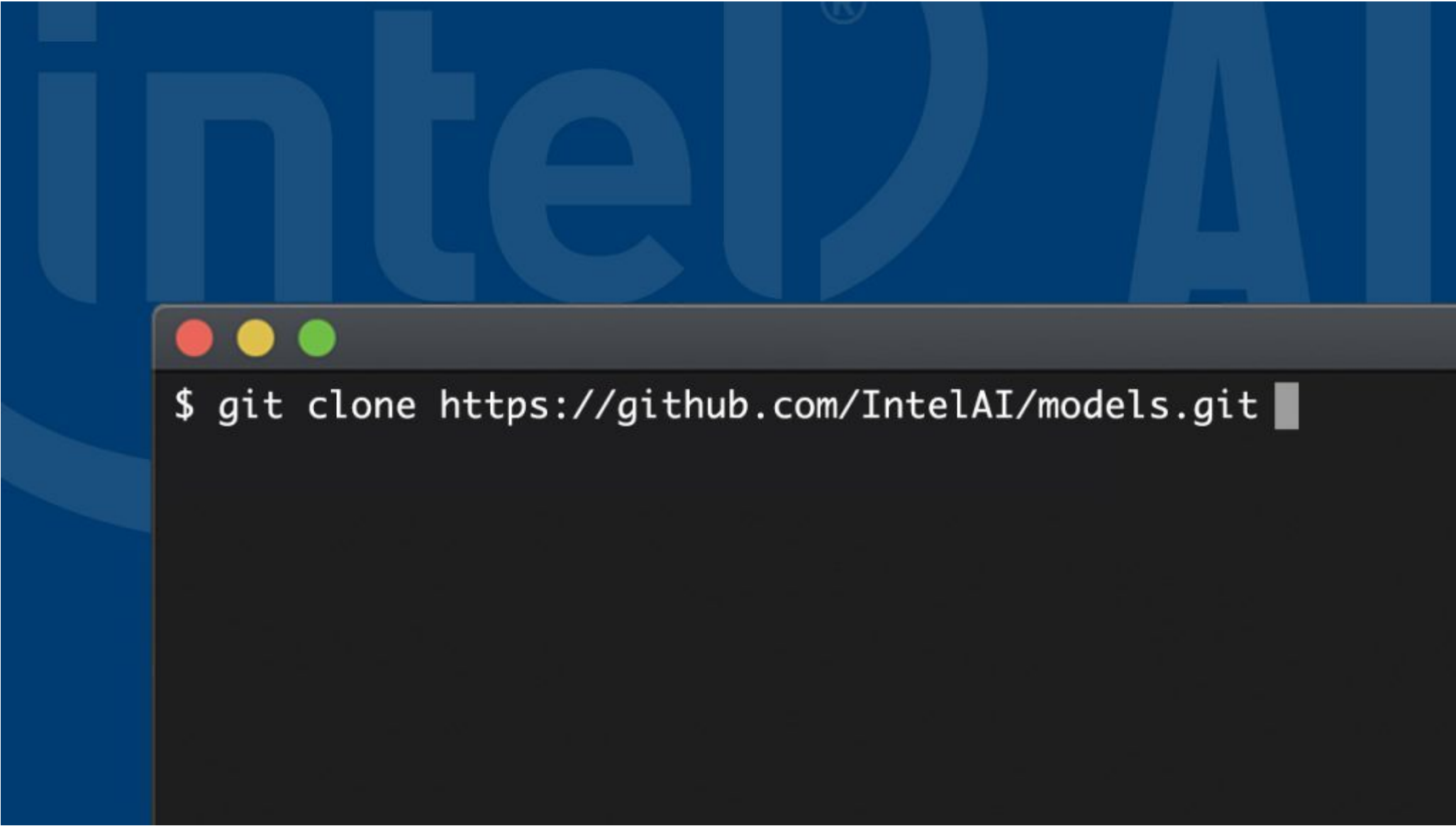
INTRODUCING THE MODEL ZOO FOR INTEL® ARCHITECTURE

Are you a data scientist who wants to optimize the performance of your machine learning (ML) inference workloads? Perhaps you've...

Read More

INTRODUCING INT8 QUANTIZATION FOR FAST CPU INFERENCE USING…

Deep learning framework optimizations and tools that streamline deployment are advancing the adoption of inference applications on Intel® platforms. Reducing…

Read More

Search AI…

Stay Connected

Keep tabs on all the latest news with our monthly newsletter.

Subscribe

@intelai   @intelairesearch   @intelai   intel-ai   intelai   nervanasystems