

Faster Machine Learning with Custom Built TensorFlow on Eagle

Paul Diaz

04/26/2021

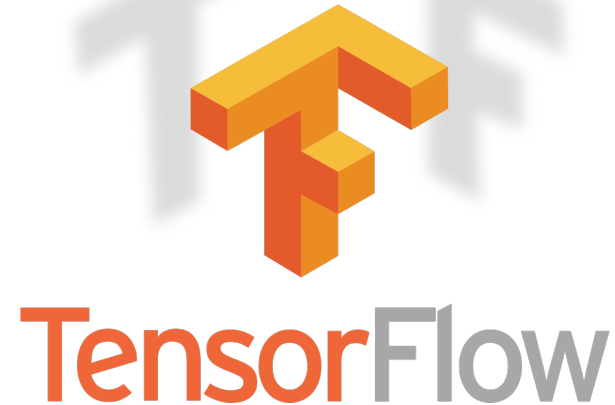
Outline

- 1** What is TensorFlow (TF)?
- 2** Optimized TF version for Eagle
- 3** How to install
- 4** How to test & benchmark TF
- 5** Available TF resources

What is TensorFlow?

A symbolic math library (fancy calculator) for machine learning

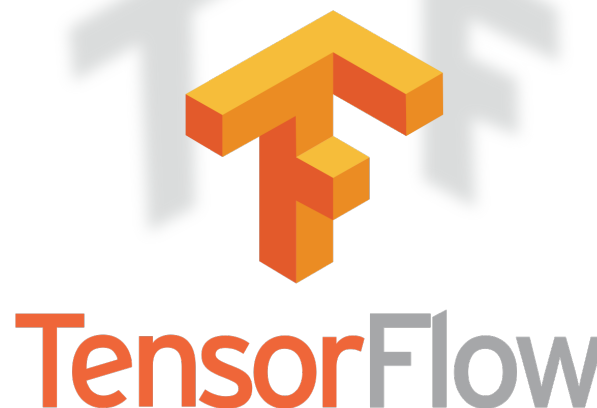
- Developed by Google in 2017
- Comparable to PyTorch (Facebook)
- Python and C APIs available for 64-bit Linux/macOS/Windows
- Without API backwards compatibility for: C++, GO, Java, and JS
- 3rd party packages for: C#, Haskell, Julia, MATLAB, R, Scala, Rust, Ocam, and Crystal



Why use TensorFlow?

TF includes build-in functionality for:

- Regression
- Clustering
- Classification
- Hidden Markov Models
- Neural Networks: Dense, Convolutional, Recurrent
- Reinforcement Learning
- Variety of activation functions, loss functions, and optimizers

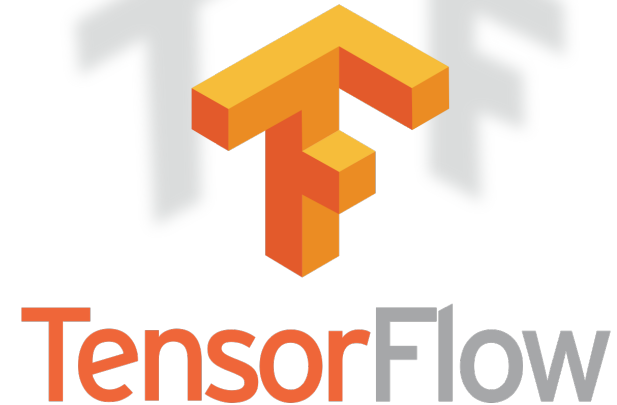


TF is designed to run on CPUs *and* CUDA enabled GPUs

Keras integration for TF 2.0+

Installing TensorFlow

Generally, TensorFlow for Python can be installed with Pip or Conda, e.g.,



Conda (with the anaconda channel):

`conda install -c anaconda tensorflow-gpu=2.2`

Works for GPU computing with proper CUDA configuration*, not optimized for CPUs

Pip:

`pip install --upgrade tensorflow==2.2.0`

Optimized for CPUs, likely not compatible with GPUs/Eagle CUDA drivers

Optimized for CPU means the TF build is made such that it utilizes the AVX512 instructions

OR....

Optimized TensorFlow Builds for Eagle

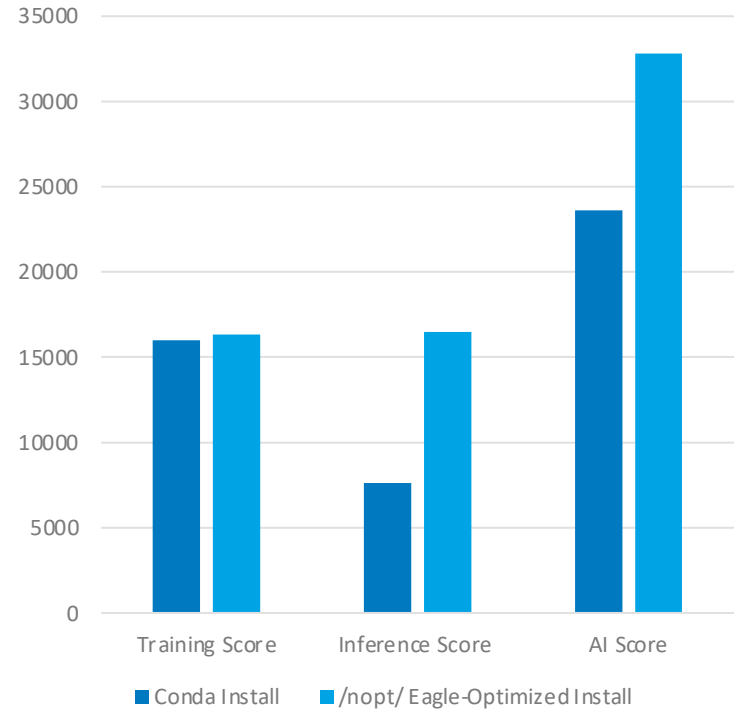
Hardware-optimized versions

- 2.0.0, 2.2.0, 2.2.1, 2.3.2, and 2.4.0
- Easily installed from Eagle /nopt/

Benefits

- Compatible with Eagle CUDA drivers
- Optimized for AVX-512 chip architecture
- GPU Enabled

GPU AI Benchmark Scores
(bigger is better!)



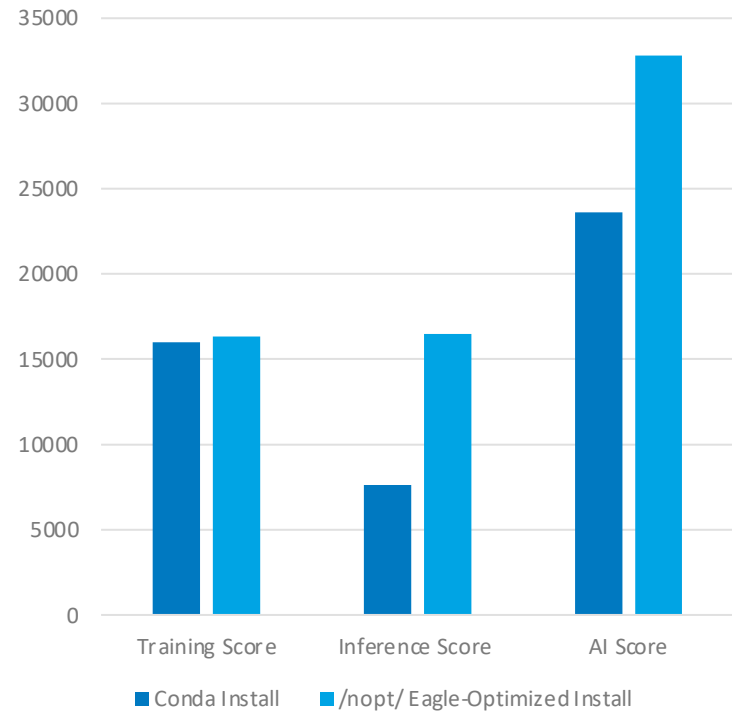
<https://pypi.org/project/ai-benchmark/>

<https://ai-benchmark.com/ranking.html>

Optimized TensorFlow Builds for Eagle

1. MobileNet-V2 [classification]
2. Inception-V3 [classification]
3. Inception-V4 [classification]
4. Inception-ResNet-V2 [classification]
5. ResNet-V2-50 [classification]
6. ResNet-V2-152 [classification]
7. VGG-16 [classification]
8. SRCNN 9-5-5 [image-to-image mapping]
9. VGG-19 [image-to-image mapping]
10. ResNet-SRGAN [image-to-image mapping]
11. ResNet-DPED [image-to-image mapping]
12. U-Net [image-to-image mapping]
13. Nvidia-SPADE [image-to-image mapping]
14. ICNet [image segmentation]
15. PSPNet [image segmentation]
16. DeepLab [image segmentation]
17. Pixel-RNN [inpainting]
18. LSTM [sentence sentiment analysis]
19. GNMT [text translation]

GPU AI Benchmark Scores
(bigger is better!)



AI Benchmark Rankings

[Phones](#) | [Mobile SoCs](#)

Deep Learning Hardware Ranking

[Desktop GPUs and CPUs](#)

[View Detailed Results](#)

Model	TF Version	Cores	Frequency, GHz	Acceleration	Platform	RAM, GB	Year	Inference Score	Training Score	AI-Score
Tesla V100 SXM2 32Gb	2.1.0	5120 (CUDA)	1.29 / 1.53	CUDA 10.1	Debian 10	32	2018	17761	18030	35791
Tesla V100 SXM2 16Gb	2.1.0	5120 (CUDA)	1.31 / 1.53	CUDA 10.1	Red Hat 7.5	16	2017	17251	17836	35086
Tesla V100 PCIe 32Gb	2.1.0	5120 (CUDA)	1.23 / 1.38	CUDA 10.1	Debian 10	32	2018	16530	17865	34394
Tesla V100 PCIe 16Gb	2.1.0	5120 (CUDA)	1.25 / 1.38	CUDA 10.1	Red Hat 7.5	16	2017	16511	17837	34347
NVIDIA Quadro GV100	1.14.0	5120 (CUDA)	1.13 / 1.63	CUDA 10	Debian 10	32	2018	16748	17132	33880

How to install optimized TF for Eagle & How to test

https://github.com/NREL/HPC/tree/master/workshops/Optimized_TF

Interested users can follow the steps outlined in the README.md file

How to benchmark

Note: the output of the benchmark script will be in a file named 'output.txt' and any errors can be viewed in 'errors.txt'

Note: The file 'TFbenchmark.sh' must be modified to include the correct modules of your target TF version, as well as the account to charge Eagle hours to.

```
cd ~/HPC/workshops/Optimized_TF/
```

```
sbatch --account=<account_name> ./TFbenchmark.sh
```

How to build from source

An example build process can be found in
`~/HPC/workshops/Optimized_TF/how_to_build_notes`

Note: This process is considerably more involved. Additional information may be found here:

<https://www.tensorflow.org/install/source>

Additional Resources

TensorFlow API Documentation:

https://www.tensorflow.org/api_docs

TensorFlow build from source & tested build configurations:

<https://www.tensorflow.org/install/source>

TensorFlow 2 intro to machine learning free course:

<https://www.youtube.com/watch?v=tPYj3fJGjk>

Eagle HPC Documentation:

<https://www.nrel.gov/hpc/eagle-system.html>

Managing Conda environments:

[Click here](#)

Q&A , Thank you!

www.nrel.gov

