

Torch7: A Matlab-like Environment for Machine Learning

Weilin Xu
01-21-2014

As the official introduction paper^[1] tells, “Torch7 is a versatile numeric computing framework and machine learning library that extends Lua”. This short report will help you experience Torch7 and teach you how to use Torch7 in your Machine Learning projects, especially Deep Learning projects. In addition, it also provides guidelines to help you upgrade your project code from Torch5 to the latest Torch7.

***Notes:** Torch7 is still under active developing. The author used the latest Torch7 in the Github master branch when writing this report.

Features

- * Lua-based
 - Ease the development of numerical algorithms
 - Could be easily extended
 - Fast
- * Parallelization capacity
 - OpenMP
 - CUDA
- * Performance overview^{[1][2]}
 - CPU-MLP: Torch7 > Theano > Matlab > NumPy >> Torch5
 - CPU-CNN: Torch7 > Theano > EBLearn > Torch5 > SciPy
 - GPU-MLP: Torch7 > Theano >> Matlab
 - GPU-CNN: Torch7 > Theano

Conclusion:

- * Don't hesitate to upgrade to Torch7 if you are using Torch5.
- * Don't waste money on Matlab if you just want to construct and train neural networks.
- * Lua beats Python, surprising?

Installation

There're at least three approaches to install Torch7.

If you just want to have a taste of Torch7, a VirtualBox image would be very helpful.

If you want to use Torch7 in your machine learning projects, you should consider the approaches 2 or 3 to install Torch7 on your host OS.

If you want to use both Torch5 and Torch7 on your machine, the approach 2 should be your painless choice.

1. VirtualBox image (Lua + Torch7)

An OVA packed Virtual Machine image is available at <http://data.neuflow.org/share/Torch7.ova>, which was created by Clement Fafrabet, one of the co-author of Torch7. The image includes a Ubuntu 12.04 OS pre-installed with Torch7.

Install VirtualBox on your machine, and double-click the OVA package, then it will be added to your VM list after a few minutes. Start the VM, open a 'terminal' program, then type the commands to run and exit Torch7:

```
$ torch
Type help() for more info
Torch 7.0 Copyright (C) 2001-2011 Idiap, NEC Labs, NYU
Lua 5.1 Copyright (C) 1994-2008 Lua.org, PUC-Rio
t7>
t7> <Ctrl+D>
Really quit [y/N]? y
$
```

***Note:** Don't press <Ctrl+Z> to exit Torch7 because it just makes torch7 run in the background and it won't free the memory. The package management program is named as 'luarocks'.

However, the image is not good for these reasons:

- * The author didn't provide the password. You would need a live CD to reset the password if you want to gain the root privilege.
- * The VirtualBox Guest Additions software was not pre-installed.
- * You may not be able to use CUDA in a virtual machine environment.

If you want to use Torch7 in your real big-data project, you should install Torch7 on a physical machine. Follow the instructions as below.

2. Install Luajit + Torch7

This approach is posted in the homepage of the Torch7 official website at <http://torch.ch>.

```
$ curl -sk https://raw.githubusercontent.com/torch/einstall/master/install-deps | bash
$ curl -sk https://raw.githubusercontent.com/torch/einstall/master/install-luajit+torch | bash
```

In this way, you should run torch7 by typing '\$ th'. The package management program should be '\$ luarocks'.

```
$ th
>
> exit
```

```
Do you really want to exit ([y]/n)? y
$
```

If you want to install both Torch5 and Torch7 on your machine, this should be the easiest one approach without any conflict.

3. Install Lua + Torch7 from the latest code at Github

I personally prefer the latest code at Github.

Please follow the instructions at <https://github.com/mzweilin/torch7-distro>. It should work on Ubuntu or Mac OS X.

The interface with Torch7 is almost the same as that in Approach 1, except the package management program has been renamed as `torch-rocks`

Tutorial examples

There're several tutorials for Torch7 online:

* Official tutorial: training a Neural Network

The tutorial and the online manual of Torch7 are available at:

<http://www.cs.virginia.edu/~wx4ed/cs6501/torch7/tutorial/index.html>

If you have experience with Torch5, you would find this tutorial is almost the same as that one in Torch5.

* Summer school program on Deep Learning

<http://code.cogbits.com/wiki/doku.php?id=start>

Tutorial 1: Setup / Basics / Getting Started

Tutorial 2: Supervised Learning

Tutorial 3: Unsupervised Learning

Tutorial 4: Graphical Models

Tutorial 5: Creating New Modules

Tutorial 6: Using CUDA

* Official demos (modified by Weilin)

<https://github.com/mzweilin/torch-demos>

Tensors

Linear Regression

Train a digit classifier

...

Changes on Torch7

The evolution sequence of Torch is listed as below:

Torch3 -> Torch5 -> **Torch7** -> Torch9
deprecated -> deprecated -> **best choice** -> beta

Torch7 is usually the best choice for your project, since Torch3 and Torch5 have been deprecated by the development team while Torch9 is still in the beta stage.

However, some recent projects may still use Torch5. It's important to understand the difference between Torch5 and Torch7, so as to update the project code to Torch7.

The significant changes in the latest Torch7 include:

1. Startup

Torch7 has a more elegant startup method, just ``$ torch`` (or ``$ th``) rather than ``$ lua`` and then `> require 'torch'`.

2. Package paths

The package paths in Torch7 have been changed a lot. For example, there's no package named 'lab' in the latest Torch7. Instead, the functions in 'lab' have been merged into the 'torch' package, which is loaded by default when starting Torch7.

Unfortunately, there's no complete documentation on such changes. Here's the changes that I met in several projects.

```
gfx -> plot -> gnuplot  
lab => torch  
sys.filep() -> paths.filep()  
...
```

3. New support on Data Types

Torch7 could switch easily between floating types (float, doubles, or CUDA)

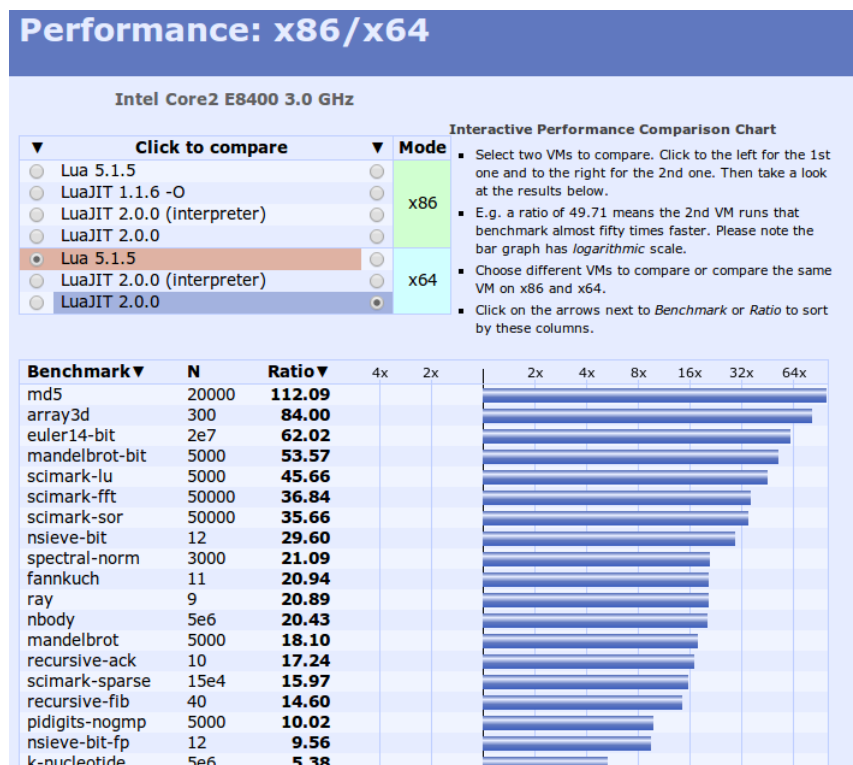
4. Parallelization capability

Torch7 has parallelization capabilities, both OpenMP and CUDA. Thanks to the good implementation in Torch7, it's almost transparent to users. You're not necessary to understand the details of OpenMP or CUDA, but you can definitely take advantage of them.

5. Luajit rather than Lua

Torch7 integrates Luajit rather than Lua, compared to Torch5. Fortunately, this change should not result in any code changes. According to the official site of Luajit, it has much better performance that gained by JIT (Just-In-Time) compilation technology. Such technology has been widely used in web browsers in order to improve the performance of JavaScript.

http://luajit.org/performance_x86.html



6. No SVM package.

Hopefully, there're not many Torch5 projects using SVM.

7. Advanced Package management

The program 'torch-rocks' makes it fairly easy to install or remove Torch7 packages on your machine.

Reference

- [1] Collobert, Ronan, Koray Kavukcuoglu, and Clément Farabet. "Torch7: A matlab-like environment for machine learning." BigLearn, NIPS Workshop. 2011.
- [2] Bergstra, James, et al. "Theano: a CPU and GPU math expression compiler." Proceedings of the Python for Scientific Computing Conference (SciPy). Vol. 4. 2010.