

# 深度学习的一些软件 对比

- 各软件简介

- **Caffe**

- 第一个主流的工业级深度学习工具。它开始于2013年底,由UC Berkely的Yangqing Jia老师编写和维护的具有出色的卷积神经网络实现。在计算机视觉领域Caffe依然是最流行的工具包。它有很多扩展，但是由于一些遗留的架构问题，不够灵活且对递归网络和语言建模的支持很差。

- **TensorFlow**

- Google开源的其第二代深度学习技术——被使用在Google搜索、图像识别以及邮箱的深度学习框架。是一个理想的RNN（递归神经网络）API和实现，TensorFlow使用了向量运算的符号图方法，使得新网络的指定变得相当容易，支持快速开发。缺点是速度慢，内存占用较大。（比如相对于Torch）

- **MXNet**

- 是李沐和陈天奇等各路英雄豪杰打造的开源深度学习框架，是分布式机器学习通用工具包DMLC 的重要组成部分。它注重灵活性和效率，文档也非常的详细，同时强调提高内存使用的效率，甚至能在智能手机上运行诸如图像识别等任务。

- **Torch**

- Facebook力推的深度学习框架，主要开发语言是C和Lua。有较好的灵活性和速度。它实现并且优化了基本的计算单元，使用者可以很简单地在此基础上实现自己的算法，不用浪费精力在计算优化上面。核心的计算单元使用C或者cuda做了很好的优化。在此基础之上，使用lua构建了常见的模型。缺点是接口为lua语言，需要一点时间来学习。

- **Theano**

- 2008年诞生于蒙特利尔理工学院，主要开发语言是Python。Theano派生出了大量深度学习Python软件包，最著名的包括Blocks和Keras。Theano的最大特点是非常的灵活，适合做学术研究的实验，且对递归网络和语言建模有较好的支持，缺点是速度较慢。

- 各软件对比

库名称	开发语言	速度	灵活性	文档	适合模型	平台	上手
Caffe	c++/cuda	快	一般	全面	CNN	所有系统	中等
TensorFlow	c++/cuda/python	中等	好	中等	CNN/RNN	Linux\OSX	难
MXNet	c++/cuda	快	好	全面	CNN	所有系统	中等
Torch	c/lua/cuda	快	好	全面	CNN/RNN	Linux\OSX	中等
Theano	python/c++/cuda	中等	好	中等	CNN/RNN	Linux\OSX	易

# Overview

	Caffe	Torch	Theano	TensorFlow
Language	C++, Python	Lua	Python	Python
Pretrained	Yes ++	Yes ++	Yes (Lasagne)	Inception
Multi-GPU: Data parallel	Yes	Yes <small>cunn. DataParallelTable</small>	Yes <small>platoon</small>	Yes
Multi-GPU: Model parallel	No	Yes <small>fbcunn.ModelParallel</small>	Experimental	Yes (best)
Readable source code	Yes (C++)	Yes (Lua)	No	No
Good at RNN	No	Mediocre	Yes	Yes (best)

Fei-Fei Li & Andrej Karpathy & Justin Johnson

Lecture 12 - 145 22 Feb 2016

斯坦福的CS231n - Convolutional Neural Networks for Visual Recognition(Winter 2016)中的Lecture 12中，由课程讲师Justin Johnson详细介绍了他个人对于主流第三方库的实践经历和看法

- 几个用例问题:
- **1.Extract AlexNet or VGG features?** Use Caffe
- **2.Fine tune AlexNet for new classes?** Use Caffe
- **3.Image caption with finetuning?**
  - -> Need pretrained models (Caffe, Torch, Lasagne)
  - -> Need RNNs (Torch or Lasagne)
  - -> Use Torch or Lasagna
- **4.Segmentation? (Classify every pixel)**
  - -> Need pretrained model (Caffe, Torch, Lasagna) -> Need funny loss function
  - -> If loss function exists in Caffe: Use Caffe
  - -> If you want to write your own loss: Use Torch

- **5.Object Detection?**

- -> Need pretrained model (Torch, Caffe, Lasagne)
- -> Need lots of custom imperative code (NOT Lasagne) -> Use Caffe + Python or Torch

- **6.Language modeling with new RNN structure?**

- -> Need easy recurrent nets (NOT Caffe, Torch)
- -> No need for pretrained models
- -> Use Theano or TensorFlow

- **7.Implemente BatchNorm?**

- -> Don't want to derive gradient? Theano or TensorFlow
- -> Implement efficient backward pass? Use Torch

# My Recommendation

Feature extraction / finetuning existing models: Use Caffe

Complex uses of pretrained models: Use Lasagne or Torch

Write your own layers: Use Torch

Crazy RNNs: Use Theano or Tensorflow

Huge model, need model parallelism: Use TensorFlow