

代码之神 Jemas Dean
非Eager 模式
知识图谱 LSTM

Tensorflow

- 数据流图 多维数组（张量） CPU GPU
- GPU 上千核
- CUDA 英伟达 数学运算
- 计算机图形学
- Caffe MxNet
- 支持 Kubernetes K8 容器部署

为什么选择tensorflow

Python 接 口

- 便捷性/灵活性:可以将计算模型部署到一个或多个桌面、服务器、移动等多种设备(CPUs or GPUs);适用于多种系统 Raspberry Pi, Android, Windows, iOS, Linux 到 serverfarms
- 可视化:了解下TensorBoard
- 可以保存/还原模型
- 自动微分(Auto-differentiation autodiff)
- 庞大社区，非常流行(>30万提交次数， >8.5万相关开源库)

深度学习框架

- Caffe – Caffe2 **facebook**
- Torch – PyTorch
- Theano – TensorFlow Google
- Keras
- CNTK Microdsoft
- MXNet Amazon
- and so on ...

Google:

TensorFlow

↑

“One framework
to rule them all”

Facebook:

PyTorch +Caffe2

↑

Research

↑

Production

引自cs231n_2017_lecture8

- PaddlePaddle baidu 百度
- TPU 芯片 英伟达

TensorFlow is a safe bet for most projects. Not perfect but has huge community, wide usage. Maybe pair with high-level wrapper (Keras, Sonnet, etc)

I think **PyTorch** is best for research. However still new, there can be rough patches.

Use **TensorFlow** for one graph over many machines

Consider **Caffe, Caffe2, or TensorFlow** for production deployment

Consider **TensorFlow** or **Caffe2** for mobile

引自cs231n_2017_lecture8

TensorFlow 基础入门

- 安装
- TensorFlow数据流图

TensorFlow数据流图

TensorFlow 是一种采用数据流图（data flow graphs），用于数值计算的开源软件库。

其中**Tensor** 代表传递的数据为**张量**（多维数组），Flow 代表使用计算图进行运算。

- 数据流图用「节点」（nodes）和「边」（edges）组成的有向图来描述数学运算。

节点一般用来表示施加的**数学操作**，但也可以表示**数据输入**的起点和**输出**的终点，或者是读取 / 写入持久变量（persistent variable）的终点。

边表示节点之间的输入 / 输出关系。这些数据边可以传送维度可动态调整的多维数据数组，即**张量**（tensor）。

- 计算图与会话
- 计算图

为了实际评估节点，必须在会话内运行计算图。简言之，编写的代码只生成仅仅用来确定 张量的预期大小以及对它们执行的运算的图。但是，它不会为任何张量赋值。

因此，TensorFlow Graph 类似于 Python 中的函数定义。它「不会」为你执行任何计 算(就像函数定义不会有任何执行结果一样)。它「仅」定义计算操作

- 会话(Session)
- TensorFlow 张量

TensorFlow 中最基本的单位是常量(Constant)、变量(Variable)和占位符(Placeholder)。

常量定义后值和维度不可变。

变量定义后值可变而维度不可变。 在神经网络中，变量一般可作为储存权重和其他信息的矩阵，而常量可用来储存超参数或其他结构信息。

占位符用来为输入数据占位。

- 常量
- 变量
- 占位符

我们已经创建了各种形式的常量和变量，但 TensorFlow 同样还支持占位符。占位符并没有初始值，它只会分配必要的内存。在会话中，占位符可以使用 feed_dict 馈送数据。

feed_dict 是一个字典，在字典中需要给出每一个用到的占位符的取值。在训练神经网络时需要每次提供一个批量的训练样本，如果每次迭代选取的数据要通过常量表示，那么 TensorFlow 的计算图会非常大。因为每增加一个常量，TensorFlow 都会在计算图中增加一个节点。所以说拥有几百万次迭代的神经网络会拥有极其庞大的计算图，而占位符却可以解决这一点，它只会拥有占位符这一个节点。

- Datasets

在初学时，我们只需要关注两个最重要的基础类:Dataset和Iterator。Dataset可以看作是相同类型“元素”的有序列表。在实际使用时，单个“元素”可以是向量，也可以是字符串、图片，甚至是tuple或者dict。

先以最简单的，Dataset的每一个元素是一个数字为例:

```
import tensorflow as tf
import numpy as np
dataset = tf.data.Dataset.from_tensor_slices(np.array([1.0, 2.0, 3.0, 4.0, 5.0]))
```

- 操纵函数 map batch shuffle repeat

直线参数估计问题

用迭代的思想去解决问题:
首先需要定义函数的形式:
y=a*x+b 之后那么我们优化的目标就是使得d和y越来越接近，衡量这种接近 用“二范数”也就是距离的平方，所有衡量接近的函数我们都称之为损失函数

MNIST数据集

- M NIST是机器学习中的经典数据集，其包括图像处理、机器分类等 多个维度的问题。

解决问题过程

-
- BATCHSIZE
- Sigmoid和二值交叉熵 分类问题
- Softmax和交叉熵
- 二分类下sigmoid和softmax等价