# quantization工具分析

## 1. 工具简介

这是tensorflow提供的量化工具，运行后会发现模型变小。TensorFlow 自带对八位运算的生产级支持。它也能把浮点模型转换为等价的使用量化计算进行推断的图。

在TensorFlow/tensorflow/tools/quantization/下有一个python脚本，即quantize\_graph.py，它的功能就是将float型的.pb文件转换为int型的.pb文件，更具体点，它分为两个步骤：

1）将所有float型的ops（包括了fake-quant）转换为quant的ops；

2）根据quant的ops的输入类型的要求，将类型名为“Const”的参数从float型转换为int型。当然，min/max仍然为float。

## 2. 具体使用

### 2.1 首先要freezon

### 2.2 融合BN的参数

将conv的参数和BN的参数merge；就是跑optimize\_for\_inference.py

### 2.3 转换

将会把.pb模型变成quan-conv的.pb模型；即跑quantization/quantize\_graph脚本 "这个文件可以把将FakeQuantWithMinMaxVars转换成Requantize，这也就表明使用fake训练后的模型，能转换成quan-conv的模型。

另外.pb也能转换成tflite的模型！！"

a）编译quantize\_graph

bazel build tensorflow/tools/quantization:quantize\_graph

b）运行quantize\_graph

INPUT=/home/ydwu/project/mobilenet-quant-Q\_DW/quantization-Q\_DW/merge\_model/merge\_const.pb

OUTPUT=/home/ydwu/project/mobilenet-quant-Q\_DW/quantization-Q\_DW/result\_model/quantization.pb

OUTPUT\_NODE\_NAMES=MobilenetV1/Predictions/Reshape\_1

CUDA\_VISIBLE\_DEVICES="" \

python \

/home/ydwu/project/mobilenet-quant-Q\_DW/quantization-Q\_DW/src/quantize\_graph.py \

--input=${INPUT} \

--output\_node\_names=${OUTPUT\_NODE\_NAMES} \

--output=${OUTPUT} \

--mode=eightbit

### 2.4 测试

运行label\_image来测试

首先导入的是weight等网络参数，当然首先量化的也是它们；然后读入图片，经过influnence过程（在这中间还会再量化weight）。

## 参考资料：

https://github.com/tensorflow/tensorflow/tree/edcfb9e7ac77ab7e6dbf524fce65fe4146721259/tensorflow/tools/graph\_transforms

http://blog.163.com/wujiaxing009@126/blog/static/71988399201741464120526?ignoreua

http://fjdu.github.io/machine/learning/2016/07/07/quantize-neural-networks-with-tensorflow.html

http://blog.csdn.net/u011961856/article/details/76736103

http://www.infoq.com/cn/articles/application-of-tensorflow-in-intelligent-terminal

https://www.cnblogs.com/powercsj/p/7580737.html

http://blog.csdn.net/qq\_40027052/article/details/78485997

https://blog.csdn.net/cokeonly/article/details/79024279

https://blog.csdn.net/leozwang/article/details/79392628