# 检测模型剪枝相关实验

作者：张松

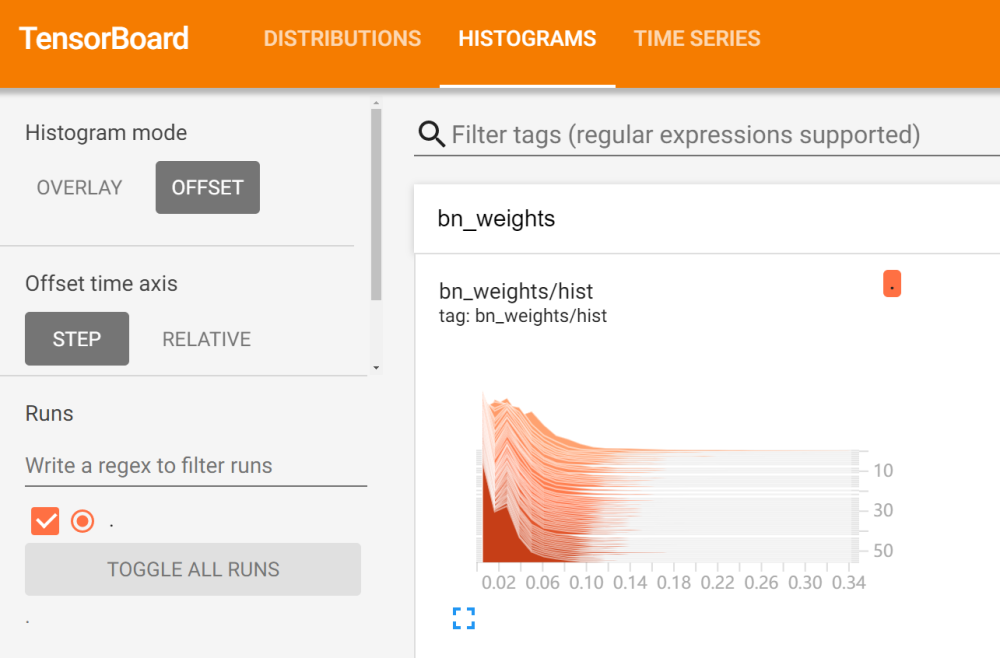
1.代码框架：mmdetection

剪枝方法:Learning Efficient Convolutional Networks Through Network Slimming论文，

剪枝对象：算力为8.4G的darknet21\_yoloneck-yolohead\_512\_192模型，map为74%

剪枝代码:10.7.0.102：/home/szhang/project/gitlab/git/mmdetection-dev/configs/yolov3\_multitask/yolov3\_tevs2d\_darknet21\_singlehead\_prunning\_zs\_sparse.py,

用稀疏率0.001做稀疏训练，稀疏训练map=0.7，bn层权重稀疏结果如下：



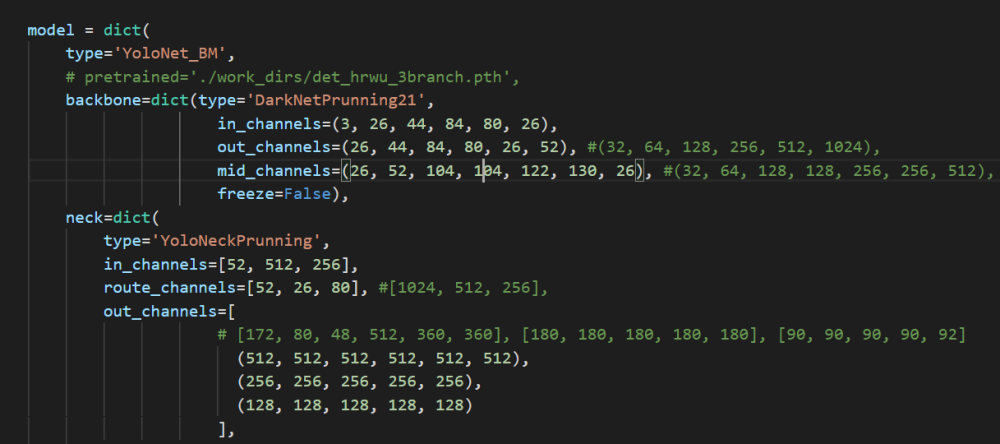
2.为了增加剪枝力度，修改剪枝算法策略，把模型裁剪到了3.94G，并进行fine-tuning训练。算法修改的策略如下：

a: 设置每层至少裁剪掉20%的通道。

b：每个resblock模块，在shortcut层输入的两个卷积裁剪通道改成求并集等算法逻辑。

剪枝代码：10.7.0.102：/home/szhang/project/gitlab/git/mmdetection-dev/tools/prunning\_toolbox

剪枝后模型各层通道情况（可以发现backbone的前后通道剪枝比较均匀）：



3.分析之前剪枝算法只剪枝backbone后面几层通道的原因，于是打印出bn层的weights权重，发现backbone的前面几层bn参数值普遍都比后面几层大，所以原因就是稀疏化训练后backbone后几层bn参数值偏小导致的。

于是打印了原始模型的bn层的weights权重，发现backbone的前面几层bn参数值同样都比后面几层大。

|  |  |
| --- | --- |
| 稀疏化训练后backbone前几层bn参数值 | 稀疏化训练后backbone后几层bn参数值 |
|  |  |

5.3.94G的模型，fine-tuning目前迭代了88个epoch，map=67.9%，与之前的模型相比，map下降了6个点。