Deep Learning 读书会第八次讨论记录（由极视角整理）

下面为2017年1月8日 Deep Learning 读书会第六章深度前馈网络6.4-6.6。本次讨论问题由@希希 整理，并组织讨论。笔记由极视角筱彤整理。如有想加入读书会讨论的，请联系小助手（微信：Extreme-Vision）.

讨论话题

话题一. 在神经网络结构设计中，我们可以用卷积来减少不同层之间的连接个数，但是是否卷积的 size 可以在训练中学习，如何来学习呢?（希希）

希希

一般神经网络的 filter 大小决定了特征提取的数据规模大小，googlenet 里面会同时使用多个 size 的 filter，但是这样也意味着参数的增加。如果对数据本身有

个大致的判断，生活中我们往往会选择一个合适的 filter size。但是有什么办法根据数据本身的特性，或者在学习的过程中使用一种带有 threshold 的函数，来决定 filter size 呢？

极视角小助手
对这个问题大家有什么看法呢？

人工智障 v1.04
是要刚好合适不多不少吗？

Michael
我感觉现在大家都在往小核发展

人工智障 v1.04
估计现在的网络结构冗余程度是比较大的

Michael
小核多层能等效为大核

希希
基本上 size 越大越好，但是大了肯定增加参数个数，也影响速度。2 个 3*3 等于 1 个 5*5。甚至效果应该还好些，但是都造成了参数增加。

枫
有没有比较小核多层和大核表达能力的论文？

极视角小助手
这个是不是就是 要选择 size 和速度的平衡点呢

希希

是的

九问

卷积的 size 可以学习么，好像不可以吧

人工智障 v1.04

目测大家都是用“经验值”吧，也没有一个很准确的理论上的解释

九问

目前都是手工调

希希

我也想过单纯把不同 size 的网络都试一遍，只是需要的时间肯定特别多

九问

而且不仅都往小的 kernal 调，现在有各种变体，像 inception，多层 filter，concat 一起

枫

其实还有个卷积核个数问题

九问

嗯，都是人工经验啊

希希

卷积核个数也是一个调试的地方，比如开始说的两个 3*3，但是卷积核减半，可能会比一个 5*5 还好

极视角小助手

所以只是一种累积的经验 并没有数学公式来总结吗

九问
对啊

希希
我觉得数学公式估计没法做，但是有没有一些经验上的指标
枫
2 个 3*3 的特征是不是更抽象，是不是能获得更多细节？

人工智障 v1.04
这也是深度学习玄学的一部分

安兴乐
不要把它当玄学，除了经验还要看对具体问题的理解。

希希
比如 patch 的方差，又或者可以通过类试 autoencoder 的方法先预估一下。autoencoder 也许对于数据本身的多样程度有种大致描绘。多样性强，变化陡峭的数据更加需要小 size，多层。反之亦然

极视角小助手
那这种经验上的指标并不是通用的吧。如果有的话

希希
不可能通用。一般也只是一个大致指示

枫
具体一般有哪些分析数据的方法，如何衡量数据的多样性和陡峭程度？

希希
不是很清楚，我想专门做图像的也许比较了解这些指标

A Yang
我觉得现在关注卷积核大小不如关注卷积数量

希希
可能 pca 也不错。我也想过，卷积数量和大小是个 tradeoff

A Yang
嗯 很多时候冗余很多
希希
不如就写个卷积数量的 0 norm 做正则项。优化的时候采用 threshold 函数。低于一定值就剪掉。轮流优化。算了，随便说说，可行性不高。

A Yang
有人实现类似的，就是把一些权重很低的去除

枫
这不是模型压缩吗？

A Yang
是的，0 范 是哪种范数？

极视角小助手
那其实我们第一个话题讨论得差不多了，可能还需要再思考后交流。我们接着第二个问题吧。

话题二. maxout 可以除以特征图数量 k，那么 maxout 减少了特征图数量，这是否意味我每一层需要更多的卷积核来保证特征图数量不至于太少，而这又是否意味着参数会变多，造成过拟合？Lightened CNN 中采用了 k 为 2 的 maxout，那么他的效果和 relu 比有什么不同?（希希）

希希
我觉得这个问题和第一个其实差不多，也是关心的是使用 maxout，后卷积核数量增加了，那么这种增加会不会影响速度，甚至造成过拟合。

极视角小助手

这个问题大家怎么理解呢

希希

lightened cnn 采用的参数比一般网络少，但是人脸识别的效果还是可以的，那么这个是否归功于 maxout 使用呢

枫

知乎上看到的

Maxout 公式如下：

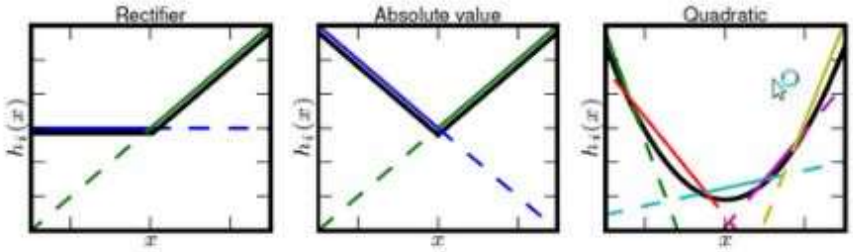
$f_i(x) = max_{j \in [1,k]} z_{ij}$

假设 w 是2维，那么有：

$f(x) = max(w_1^T x + b_1, w_2^T x + b_2)$

可以注意到，ReLU 和 Leaky ReLU 都是它的一个变形（比如，$w_1, b_1 = 0$ 的时候，就是 ReLU）.

Maxout的拟合能力是非常强的，它可以拟合任意的的凸函数。作者从数学的角度上也证明了这个结论，即只需2个maxout节点就可以拟合任意的凸函数了（相减），前提是"隐隐含层"节点的个数可以任意多.



所以，Maxout 具有 ReLU 的**优点**（如：计算简单，不会 saturation），同时又没有 ReLU 的一些缺点（如：容易 go die）。不过呢，还是有一些**缺点**的嘛：就是把参数double了。

希希

Maxout 拟合是采用线性逼近的方法，也就是使用多条直线逼近曲线，那么由于曲线可以用直线拟合，这就需要考虑曲线梯度的变化，也就是二阶导，一般，二

阶导小，意味着直线的个数少，也就是 k 使用的少，二阶导大意味着 k 多，那么 k 与卷积核的个数成正比，也就意味着我们希望尽可能使得 k 小，所以这个需要对拟合函数有个大致的认知。

极视角小助手

那为什么会出现过拟合呢

希希

参数多自然就过拟合

极视角小助手

那其实这个问题和第一个一样也感觉是要找个平衡点。。。那 maxout 和 relu 的效果方面又怎么说呢

AG-GROUP 元芳 第四章翻译

一般小样本就会造成 overfitting，没有一个很合理的解释

希希

http://www.cnblogs.com/tornadomeet/p/3428843.html

我不确定这篇文章是用哪些激活函数去比较的

极视角小助手

额 sigmoid 函数吗？

希希

不太可能吧，那个不是大家都不爱用了吗

极视角小助手

好吧~大家对第二个问题好像都没有什么其他想法，那我们讨论最后一个问题吧

话题三. 后馈计算中，tensorflow，caffe 针对每个层写了对应的后馈公式，那么如果在多 gpu 计算中，是否可以通过将计算拆分来增加后馈计算的速度?（希希）

枫

据我所知 caffe 就是这么干的 caffe 还将每一次前馈计算的中间值保留用于加速

极视角小助手

那 tf 呢？

希希

这个我是完全不了解，因为我没了解过多 gpu 计算要注意的一些东西，我只是考虑后馈计算时觉得我们在计算某个 layer 的时候，每个神经元的梯度计算都可以不依赖其他神经元的梯度计算

希希

所以完全可以把上一层 layer 的梯度，中间的权重放到每个 gpu，然后算好这一层的梯度再汇总，tf 的做法没有考虑这些，tf 完全只是把 batch 均分

枫

应该也是 google 那么牛这种事情还是会做的 但两者底层的计算单元好像有点不一样 如果不是自己写框架没必要弄这么细

希希

然后每个 gpu 算其中一部分的梯度，最后相加，tf 在 cifar10 的 multigpu 教材里只是将数据分开了，但是每个数据的所有神经元的梯度还是都必须计算

极视角小助手

所以后馈计算中，多 gpu 的 caffe 要比 tf 快吗？

希希

我不是很清楚，只是 caffe 总体比 tf 快 6 倍，我没找到别人的实验结果，自己做过实验是这样的

极视角小助手

哦哦 那其实这个问题对于 caffe 来说是肯定的，但是 tf 可能还有些争议，那大家还有其他的看法吗~@安兴乐 @枫

希希

嗯，tf 源代码好像不全，api 里面没提过其他的设置来设置如何分配每一层的梯度计算分配，唯一有的是 distribute 计算，但是我比较关注多 gpu

枫

tf 没用过 但也是 g 家的东西应该不错

希希

那 caffe 是如何分配的呢？每一次后馈都吧每一层的梯度计算平均分配给每个 gpu 吗？如果有某个 gpu 先算完也不会分配更多吗？

安兴乐

tensorflow 的卷积被分到了很底层的位置，需要编译源码后才能看到

希希

我手头上没找到源码

枫

多 gpu 好像有专门研究这个的 好像是分布式机器学习

希希

哦，开始找到了，是的，但是 distribute 的那个好像不是多 gpu 的，我没看太明白。那个好像是多电脑的。

枫

你把数据分开了就相当于一种分布式的计算 就需要交互的呀

希希

我回头再看下

————————————————————**End**————————————————————

写在最后

非常感谢此次进行讨论交流的朋友们以及群内支持的朋友们 ,希望我们读书会能让大家学到更多，并且讨论后可以对原书有更独到的理解。 如有愿意提交自己对这本书的理解的，欢迎联系小助手，加入此项目组。

#广告时间#
视觉前沿资讯，将算法放至极市关注请关注极市平台公众号。