[训练深度神经网络的时候需要注意的一些小技巧](http://blog.csdn.net/silence1214/article/details/44038805)

2015-03-03 15:27 163人阅读 [评论](http://blog.csdn.net/silence1214/article/details/44038805#comments)(0) [收藏](javascript:void(0);) [举报](http://blog.csdn.net/silence1214/article/details/44038805#report)

翻译网上的哈，我觉得有很大一部分从没看到过，所以就翻译了下，如有不对的地方，欢迎指正：

1：准备数据：务必保证有大量、高质量并且带有干净标签的数据，没有如此的数据，学习是不可能的

2：预处理：这个不多说，就是0均值和1方差化

3：minibatch：建议值128,1最好，但是效率不高，但是千万不要用过大的数值，否则很容易过拟合

4：梯度归一化：其实就是计算出来梯度之后，要除以minibatch的数量。这个不多解释

5：下面主要集中说下学习率

5.1：总的来说是用一个一般的学习率开始，然后逐渐的减小它

5.2：一个建议值是0.1，适用于很多NN的问题，一般倾向于小一点。

5.3：一个对于调度学习率的建议：如果在验证集上性能不再增加就让学习率除以2或者5，然后继续，学习率会一直变得很小，到最后就可以停止训练了。

5.4：很多人用的一个设计学习率的原则就是监测一个比率（每次更新梯度的norm除以当前weight的norm），如果这个比率在10-3附近，如果小于这个值，学习会很慢，如果大于这个值，那么学习很不稳定，由此会带来失败。

6：使用验证集，可以知道什么时候开始降低学习率，和什么时候停止训练。

7：关于对weight初始化的选择的一些建议：

7.1：如果你很懒，直接用0.02\*randn(num\_params)来初始化，当然别的值你也可以去尝试

7.2：如果上面那个不太好使，那么久依次初始化每一个weight矩阵用init\_scale / sqrt(layer\_width) \* randn,init\_scale可以被设置为0.1或者1

7.3：初始化参数对结果的影响至关重要，要引起重视。

7.4：在深度网络中，随机初始化权重，使用SGD的话一般处理的都不好，这是因为初始化的权重太小了。这种情况下对于浅层网络有效，但是当足够深的时候就不行了，因为weight更新的时候，是靠很多weight相乘的，越乘越小，有点类似梯度消失的意思（这句话是我加的）

8：如果训练RNN或者LSTM，务必保证gradient的norm被约束在15或者5（前提还是要先归一化gradient），这一点在RNN和LSTM中很重要。

9：检查下梯度，如果是你自己计算的梯度。

10：如果使用LSTM来解决长时依赖的问题，记得初始化bias的时候要大一点

12：尽可能想办法多的扩增训练数据，如果使用的是图像数据，不妨对图像做一点扭转啊之类的，来扩充数据训练集合。

13：使用dropout

14：评价最终结果的时候，多做几次，然后平均一下他们的结果。