# 关于batch size的思考和实验

在“电影评论情感分析”实验中，分别就不同的batch size做了对比实验，发现batch size对最后的预测准确度有比较大的影响，具体数据如下（以预测“this is very bad”为例）：

|  |  |  |
| --- | --- | --- |
| batch size | 正面概率 | 负面概率 |
| 128 | 0.40658 | 0.59342 |
| 64 | 0.23525 | 0.76475 |
| 32 | 0.16182 | 0.83818 |

上面最终的结果batch\_size=32已经体现在了notebook中了，同时也可以看到额外增加的两个测试评论也获得了不错的结果。

那么，为什么batch size会有这么大的影响呢？在回酒店的路上和小白聊起来，深受启发，有如下的几点心得：

* batch size的大小决定了一轮迭代更新多少次参数，即进行多少次的梯度下降计算。神经网络只有在吃完一个batch size的样本后才开始梯度下降的计算。
* 设置较大的batch size可以充分发挥GPU和大内存的作用，使得迭代的速度明显加快。
* 设置较小的batch size需要更多的迭代才能完成模型的训练。
* 较大的batch size可能导致陷入“局部最优解”。以爬山为例，如果我们在一个较大的范围内环顾四周，可能看到多个局部最优解，因此就可能本着其中一个而去导致失去了探索其他最优解的机会；相反，如果batch size较小，则在一个较小的范围内可能只有一个最优解，这样我们就有机会探索尽可能多的局部最优解，找到全局最优解的机会更大。

选取多大的batch size是合适的？下面的两篇论文可能有帮助：

* [On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima](https://arxiv.org/abs/1609.04836)
* [Revisiting Small Batch Training for Deep Neural Networks](https://arxiv.org/abs/1804.07612)

# CNN在LNP中的惊艳应用

CNN在图像处理中的应用是自然而然的，在LNP中的应用确实开拓了眼界，尤其是长条形的卷积核真的很富有想象力！可能，深度学习比拼的是想象力，而不是编码。

# 理解embedding

embedding层的用意是将输入文本转换为词向量，这样无论多长的句子都可以通过固定维度的词向量来表达，以利于其他的层处理。

在LNP中embedding的应用非常广泛，几乎所有的输入都首先经过一个embedding层进行预处理。

宿宝臣

欢迎大家造访我的blog: https://subaochen.github.io