

谈谈Tensorflow的dropout



作者 Traphix (/u/5d61d3f621c4) [+ 关注](#)

2016.07.05 17:30* 字数 991 阅读 2225 评论 4 喜欢 6 阅读 2225 评论 4 喜欢 6 (/u/5d61d3f621c4)

Dropout这个概念已经推出4年了，它的详细描述见论文 (<http://arxiv.org/abs/1207.0580>)。可是呢，它仿佛是个犹抱琵琶半遮面的美女，难以捉摸！！许多文献都对dropout有过描述，但解释的含糊不清，这里呢，我也不打算解释清楚，只是通过tensorflow来看一看dropout的运行机理。
文章分两部分，第一部分介绍tensorflow中的dropout函数，第二部分是我的思考

一、tf.nn.dropout函数

首先看官方函数定义：

```
def dropout(x, keep_prob, noise_shape=None, seed=None, name=None)
```

输入是：

- x, 你自己的训练、测试数据等
- keep_prob, dropout概率
-, 其它参数不咋用，不介绍了

输出是：

- A Tensor of the same shape of x

然后我们看看官方API是怎么说这个函数的：

```
With probability keep_prob, outputs the input element scaled up by 1 / keep_prob, otherwise outputs 0. The scaling is so that the expected sum is unchanged.
```

注意，输出的非0元素是原来的“1/keep_prob” 倍！说了这么多，下面给一个程序例子：

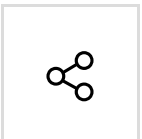
```
import tensorflow as tf

dropout = tf.placeholder(tf.float32)
x = tf.Variable(tf.ones([10, 10]))
y = tf.nn.dropout(x, dropout)

init = tf.initialize_all_variables()
sess = tf.Session()
sess.run(init)

print sess.run(y, feed_dict = {dropout: 0.4})
```

运行的结果如下：



```
[[ 0.    0.    2.5  2.5  0.    0.    2.5  2.5  2.5  2.5]
 [ 0.    2.5  2.5  2.5  2.5  2.5  0.    2.5  0.    2.5]
 [ 2.5  0.    0.    2.5  0.    0.    2.5  0.    2.5  0. ]
 [ 0.    2.5  2.5  2.5  2.5  0.    0.    2.5  0.    2.5]
 [ 0.    0.    0.    0.    0.    0.    0.    0.    2.5  2.5]
 [ 2.5  2.5  2.5  0.    2.5  0.    0.    2.5  2.5  2.5]
 [ 0.    2.5  2.5  2.5  0.    2.5  2.5  0.    0.    0. ]
 [ 0.    2.5  0.    2.5  0.    0.    2.5  2.5  0.    0. ]
 [ 2.5  2.5  2.5  2.5  2.5  0.    0.    2.5  0.    0. ]
 [ 2.5  0.    0.    0.    0.    0.    2.5  2.5  0.    2.5]]
```

分析一下运行结果：

- 输入和输出的tensor的shape果然是一样的
- 不是0的元素都变成了原来的“1/keep_prob” 倍

特点分析完毕，小总结一下， dropout这个概念看起来好高大上，然而在程序中实现竟然如此简单！说白了， tensorflow中的dropout就是：**使输入tensor中某些元素变为0，其它没变0的元素变为原来的1/keep_prob大小！**

二、关于dropout的吐槽

首先引用此博文 (<http://blog.csdn.net/hjimce/article/details/50413257>)的话：

个人总结：个人感觉除非是大型网络，才采用dropout，不然我感觉自己在一些小型网络上，训练好像很是不爽。之前搞一个比较小的网络，搞人脸特征点定位的时候，因为训练数据不够，怕过拟合，于是就采用dropout，最后感觉好像训练速度好慢，从此就对dropout有了偏见，感觉训练过程一直在波动，很是不爽。

然后，我就自己试了试，看看小型网络中dropout效果到底怎么样，程序片段如下：

```
def inference(img, dropout=1.0):
    fc1 = activation(tf.nn.bias_add(tf.matmul(img, W_fc1), b_fc1))
    # dropout
    fc1_dropout = tf.nn.dropout(fc1, dropout)
    fc2 = tf.nn.bias_add(tf.matmul(fc1_dropout, W_fc2), b_fc2)
    return fc2

cross_entropy = tf.nn.softmax_cross_entropy_with_logits(inference(X, dropout), y_)
train_step = tf.train.AdamOptimizer(1e-4).minimize(cross_entropy)
```

网络很简单，形如 784-30-10 的一个网络，只不过在输出层前用dropout处理了一下，训练的数据是MNIST的手写数据集，然后你猜怎么着？采用dropout以后的训练精度**不升反降**，后来我把网络隐藏层改成100个神经元，结果依旧，看来，我的网络还是太小了，真的如上面那篇博客所说， dropout用不好的话，真是个累赘！



Traphix (/u/5d61d3f621c4)

写了 21216 字，被 135 人关注，获得了 117 个喜欢

写了 21216 字，被 135 人关注，获得了 117 个喜欢

+ 关注

很忙很忙，或许以后还有机会再拾起深度学习



(/sign_in)发表评论

4条评论

只看作者

按喜欢排序 按时间正序 按时间倒序



6ea293ced2f3 (/u/6ea293ced2f3)

2楼 · 2017.01.09 16:25

(/u/6ea293ced2f3)

keep_prob: A scalar Tensor with the same type as x. The probability that each element is kept.

已经更新，保留element的比例。



赞



回复

Traphix (/u/5d61d3f621c4): 最近比较忙，没关注TensorFlow了。
你的意思是keep_prob的官方说明已经更新？

2017.01.09 17:07 回复



添加新评论



Yinzo (/u/b713926d927f)

3楼 · 2017.02.24 18:09

(/u/b713926d927f)

我这边也有个不是很大的网络，dropout 的确 降低了训练的速度，我用的 p=0.5 的 dropout 使得训练迭代次数多了一倍多才达到之前相同的效果，但是继续迭代之后，dropout 确实提高了一些精确度， 并且迭代次数很高也仍然没有过拟合， validation acc 大部分比 train acc 还要高



赞



回复



xuewei4d (/u/a9b9b03118b0)

4楼 · 2017.03.01 06:37

(/u/a9b9b03118b0)

dropout加不加要看你的训练是不是overfitting，不是看你“怕overfitting”



赞



回复