

reference(<https://zhuanlan.zhihu.com/p/47749934>)

## pytorch搭建模型的一些tricks

1.对两个variable进行concat操作，按道理实现方式是`c = torch.cat([a, b], dim=0)`，但提示错误：

```
TypeError: cat received an invalid combination of arguments - got (tuple, int), but expected one
(sequence[torch.cuda.FloatTensor] tensors)
(sequence[torch.cuda.FloatTensor] tensors, int dim)
didn't match because some of the arguments have invalid types: (tuple, int)
```

解决办法：根据提示刚开始以为是cat不接受tuple作为输入，然而真正的问题在于a和b的type不一样，比如可能出现a是torch.cuda.DoubleTensor而b是torch.cuda.FloatTensor，因此，将a和b转换为相同的type即可。

## 2.模型训练时提示 RuntimeError: tensors are on different GPUs

这个问题出现的原因在于训练数据data或者模型model其中有一个是\*.cuda()，而另一个不是。全都改为data.cuda()和model.cuda()即可

解决办法：

```
data = data.cuda()
model = model.cuda()
```

## 3.模型训练时提示 TypeError: argument 0 is not a Variable

原因在于输入data不是Variable，需转化成Variable格式。

解决办法：

```
from torch.autograd import Variable
data = Variable(data).cuda()
```

## 4. 模型训练时提示 RuntimeError: Trying to backward through the graph a second time, but the buffers have already been

## freed. Specify retain\_graph=True when calling backward the first time.

该问题是指在默认情况下，网络在反向传播中不允许多个backward()。需要在第一个backward设置retain\_graph=True

解决办法：

```
loss.backward()改为loss.backward(retain_graph=True)
```

## 5. 模型训练时提示 RuntimeError: multi-target not supported at

其标签必须为0~n-1，而且必须为1维的，如果设置标签为[nx1]的，则也会出现以上错误。

解决办法：

```
# print(outputs.size()) #(128L, 2L)
# print(trg)             #(128L, 1L)
loss = criterion(outputs, trg.squeeze())
```

更改为

```
# print(outputs.size()) # (128L, 2L)
# print(trg.squeeze()) # (128L, ) 或 [128]
loss = criterion(outputs, trg.squeeze())
```