TensorFlow

维度变换

主讲人：龙良曲

Outline

- shape, ndim
- reshape

- expand_dims/squeeze
- transpose
- broadcast_to

0000000000001000000000000101000 1111111111101000111111111110100 1111111111101110000000000001010 1111111111111010000000000011110

View

- [b, 28, 28]
- →[b, 28*28]
- →[b, 2, 14*28] **把图片分成上下两部分**
- →[b, 28, 28, 1]

Reshape

改变view，不改变content

```
In [80]: a=tf.random.normal([4,28,28,3])
Out[81]: TensorShape([4, 28, 28, 3])

In [82]: a.shape, a.ndim
Out[82]: (TensorShape([4, 28, 28, 3]), 4)

In [83]: tf.reshape(a,[4,784,3]).shape
Out[83]: TensorShape([4, 784, 3])

In [84]: tf.reshape(a,[4,-1,3]).shape
Out[84]: TensorShape([4, 784, 3])

In [85]: tf.reshape(a,[4,784*3]).shape
Out[85]: TensorShape([4, 2352])

In [86]: tf.reshape(a,[4,-1]).shape
Out[86]: TensorShape([4, 2352])
```

等同于上一种，-1相当于一个占位符，会自动计算替换为784

Reshape is flexible

```
In [80]: a=tf.random.normal([4,28,28,3])
Out[81]: TensorShape([4, 28, 28, 3])

In [87]: tf.reshape(tf.reshape(a,[4,-1]),[4,28,28,3]).shape
Out[87]: TensorShape([4, 28, 28, 3])

In [88]: tf.reshape(tf.reshape(a,[4,-1]),[4,14,56,3]).shape
Out[88]: TensorShape([4, 14, 56, 3])

In [89]: tf.reshape(tf.reshape(a,[4,-1]),[4,1,784,3]).shape
Out[89]: TensorShape([4, 1, 784, 3])
```

Reshape could lead to potential bugs!

- images: [4, 28, 28, 3]
  - [b, h, w, 3]

- reshape to: [4, 784, 3]
  - [b, pixel, 3]

- [4, 784, 3] $\xrightarrow{height: 28, width: 28}$ [4, 28, 28, 3]
- [4, 784, 3] $\xrightarrow{height: 14, width: 56}$ [4, 14, 56, 3]
- [4, 784, 3] $\xrightarrow{width: 28, height: 28}$ [4, 28, 28, 3]

tf.transpose

```
In [93]: a=tf.random.normal((4,3,2,1))

In [94]: a.shape
Out[94]: TensorShape([4, 3, 2, 1])

In [95]: tf.transpose(a).shape
Out[95]: TensorShape([1, 2, 3, 4])

In [97]: tf.transpose(a,perm=[0,1,3,2]).shape
Out[97]: TensorShape([4, 3, 1, 2])
```

→[b, 3, h, w]（PyTorch存储格式）

```
In [98]: a=tf.random.normal([4,28,28,3])

In [99]: tf.transpose(a,[0,2,1,3]).shape
Out[99]: TensorShape([4, 28, 28, 3])

In [101]: tf.transpose(a,[0,3,2,1]).shape
Out[101]: TensorShape([4, 3, 28, 28])

In [102]: tf.transpose(a,[0,3,1,2]).shape
Out[102]: TensorShape([4, 3, 28, 28])
```

Squeeze VS Expand_dims

Expand dim

- a: [classes, students, classes]
  - $[4, 35, 8]$
- add school dim
- [1, 4, 35, 8] + [1, 4, 35, 8]
  - $[2, 4, 35, 8]$

```
In [103]: a=tf.random.normal([4,35,8])

In [105]: tf.expand_dims(a,axis=0).shape
Out[105]: TensorShape([1, 4, 35, 8])

In [106]: tf.expand_dims(a,axis=3).shape
Out[106]: TensorShape([4, 35, 8, 1])
```

axis

axis>0，在axis前增加一个维度

axis<0，在axis后增加一个维度

```
In [103]: a=tf.random.normal([4,35,8])
In [105]: tf.expand_dims(a,axis=0).shape
Out[105]: TensorShape([1, 4, 35, 8])

In [106]: tf.expand_dims(a,axis=3).shape
Out[106]: TensorShape([4, 35, 8, 1])
In [107]: tf.expand_dims(a,axis=-1).shape
Out[107]: TensorShape([4, 35, 8, 1])

In [108]: tf.expand_dims(a,axis=-4).shape
Out[108]: TensorShape([1, 4, 35, 8])
```

Squeeze dim

- Only squeeze for shape=1 dim

- [4, 35, 8, 1]
- [1, 4, 35, 8]
- [1, 4, 35, 1]

```
In [115]: tf.squeeze(tf.zeros([1,2,1,1,3])).shape
Out[115]: TensorShape([2, 3])

In [116]: a=tf.zeros([1,2,1,3])

In [117]: tf.squeeze(a,axis=0).shape
Out[117]: TensorShape([2, 1, 3])

In [118]: tf.squeeze(a,axis=2).shape
Out[118]: TensorShape([1, 2, 3])

In [119]: tf.squeeze(a,axis=-2).shape
Out[119]: TensorShape([1, 2, 3])

In [120]: tf.squeeze(a,axis=-4).shape
Out[120]: TensorShape([2, 1, 3])
```

下一课时

Broadcasting

Thank You.