TensorFlow

合并与分割

主讲：龙良曲

Merge and split

- tf.concat

- tf.split

- tf.stack

- tf.unstack

concat

- Statistics about scores
  - [class1-4, students, scores]
  - [class5-6, students, scores]

```
In [3]: a=tf.ones([4,35,8])
In [4]: b=tf.ones([2,35,8])

In [6]: c=tf.concat([a,b],axis=0)
In [7]: c.shape
Out[7]: TensorShape([6, 35, 8])

In [8]: a=tf.ones([4,32,8])
In [9]: b=tf.ones([4,3,8])

In [10]: tf.concat([a,b],axis=1).shape
Out[10]: TensorShape([4, 35, 8])
```

Along distinct dim/axis

- *Dim=d*

df1

	A	B	C	D
0	A0	B0	C0	D0
1	A1	B1	C1	D1
2	A2	B2	C2	D2
3	A3	B3	C3	D3

df2

	A	B	C	D
4	A4	B4	C4	D4
5	A5	B5	C5	D5
6	A6	B6	C6	D6
7	A7	B7	C7	D7

df3

	A	B	C	D
8	A8	B8	C8	D8
9	A9	B9	C9	D9
10	A10	B10	C10	D10
11	A11	B11	C11	D11

Result

	A	B	C	D
0	A0	B0	C0	D0
1	A1	B1	C1	D1
2	A2	B2	C2	D2
3	A3	B3	C3	D3
4	A4	B4	C4	D4
5	A5	B5	C5	D5
6	A6	B6	C6	D6
7	A7	B7	C7	D7
8	A8	B8	C8	D8
9	A9	B9	C9	D9
10	A10	B10	C10	D10
11	A11	B11	C11	D11

df1

	A	B	C	D
0	A0	B0	C0	D0
1	A1	B1	C1	D1
2	A2	B2	C2	D2
3	A3	B3	C3	D3

df4

	B	D	F
2	B2	D2	F2
3	B3	D3	F3
6	B6	D6	F6
7	B7	D7	F7

Result

	A	B	C	D	B	D	F
0	A0	B0	C0	D0	NaN	NaN	NaN
1	A1	B1	C1	D1	NaN	NaN	NaN
2	A2	B2	C2	D2	B2	D2	F2
3	A3	B3	C3	D3	B3	D3	F3
6	NaN	NaN	NaN	NaN	B6	D6	F6
7	NaN	NaN	NaN	NaN	B7	D7	F7

stack: create new dim

- Statistics about scores
  - School1:*[classes, students, scores]*
  - School2:*[classes, students, scores]*
  - *[schools, classes, students, scores]*

```
In [19]: a.shape
Out[19]: TensorShape([4, 35, 8])
In [23]: b.shape
Out[23]: TensorShape([4, 35, 8])

In [20]: tf.concat([a,b],axis=-1).shape
Out[20]: TensorShape([4, 35, 16])

In [21]: tf.stack([a,b],axis=0).shape
Out[21]: TensorShape([2, 4, 35, 8])

In [22]: tf.stack([a,b],axis=3).shape
Out[22]: TensorShape([4, 35, 8, 2])
```

Dim mismatch

```
In [24]: a=tf.ones([4,35,8])
In [25]: b=tf.ones([3,33,8])

In [26]: tf.concat([a,b],axis=0)
InvalidArgumentError: ConcatOp : Dimensions of inputs should match: shape[0] =
[4,35,8] vs. shape[1] = [3,33,8] [Op:ConcatV2] name: concat

In [27]: b=tf.ones([2,35,8])
In [28]: c=tf.concat([a,b],axis=0)

In [29]: tf.stack([a,b],axis=0)
InvalidArgumentError: Shapes of all inputs must match: values[0].shape = [4,35,8]
!= values[1].shape = [2,35,8] [Op:Pack] name: stack
```

Unstack

```
In [30]: a.shape # TensorShape([4, 35, 8])
In [32]: b=tf.ones([4,35,8])

In [33]: c=tf.stack([a,b])
In [34]: c.shape
Out[34]: TensorShape([2, 4, 35, 8])

In [35]: aa,bb=tf.unstack(c,axis=0)
In [36]: aa.shape,bb.shape
Out[36]: (TensorShape([4, 35, 8]), TensorShape([4, 35, 8]))

# 【2, 4, 35, 8】
In [41]: res=tf.unstack(c,axis=3)

In [42]: res[0].shape, res[7].shape
Out[42]: (TensorShape([2, 4, 35]), TensorShape([2, 4, 35]))
```

Split

- VS unstack

```
# [2, 4, 35, 8]
In [43]: res=tf.unstack(c,axis=3)
In [44]: len(res)
Out[44]: 8

In [45]: res=tf.split(c,axis=3, num_or_size_splits=2)
In [46]: len(res)
Out[46]: 2

In [47]: res[0].shape
Out[47]: TensorShape([2, 4, 35, 4])

In [48]: res=tf.split(c,axis=3, num_or_size_splits=[2,2,4])

In [49]: res[0].shape, res[2].shape
Out[49]: (TensorShape([2, 4, 35, 2]), TensorShape([2, 4, 35, 4]))
```

下一课时

数据统计

Thank You.