tf.expand_dims

```
expand_dims(
    input,
    axis=None,
    name=None,
    dim=None
)
```

Defined in `tensorflow/python/ops/array_ops.py` .

See the guide: Tensor Transformations > Shapes and Shaping

Inserts a dimension of 1 into a tensor's shape.

Given a tensor `input` , this operation inserts a dimension of 1 at the dimension index `axis` of `input` 's shape. The dimension index `axis` starts at zero; if you specify a negative number for `axis` it is counted backward from the end.

This operation is useful if you want to add a batch dimension to a single element. For example, if you have a single image of shape `[height, width, channels]` , you can make it a batch of 1 image with `expand_dims(image, 0)` , which will make the shape `[1, height, width, channels]` .

Other examples:

```
# 't' is a tensor of shape [2]
tf.shape(tf.expand_dims(t, 0))  # [1, 2]
tf.shape(tf.expand_dims(t, 1))  # [2, 1]
tf.shape(tf.expand_dims(t, -1))  # [2, 1]

# 't2' is a tensor of shape [2, 3, 5]
tf.shape(tf.expand_dims(t2, 0))  # [1, 2, 3, 5]
tf.shape(tf.expand_dims(t2, 2))  # [2, 3, 1, 5]
tf.shape(tf.expand_dims(t2, 3))  # [2, 3, 5, 1]
```

This operation requires that:

`-1-input.dims() <= dim <= input.dims()`

This operation is related to `squeeze()` , which removes dimensions of size 1.

Args:

- `input` : A `Tensor` .
- `axis` : 0-D (scalar). Specifies the dimension index at which to expand the shape of `input` . Must be in the range `[-rank(input) - 1, rank(input)]` .
- `name` : The name of the output `Tensor` .
- `dim` : 0-D (scalar). Equivalent to `axis` , to be deprecated.

Returns:

A `Tensor` with the same data as `input` , but its shape has an additional dimension of size 1 added.

Raises:

- `ValueError` : if both `dim` and `axis` are specified.

Last updated November 2, 2017.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**

Raises:

- `ValueError` : if both `dim` and `axis` are specified.

Last updated November 2, 2017.