

tf.expand_dims



[TensorFlow](#)
[1 version](#) [\(/versions/r1.15/api_docs/python/tf/expand_dims\)](#)



[View](#)
[source](#) (https://github.com/tensorflow/tensorflow/blob/master/tensorflow/python/ops/stack_ops.py#L436)
[on](#) [GitHub](#)

Returns a tensor with a length 1 axis inserted at index `axis`.

```
tf.expand_dims(  
    input, axis, name=None  
)
```

Used in the notebooks

Used in the guide	Used in the tutorials
<ul style="list-style-type: none">• Masking and padding with Keras (https://www.tensorflow.org/guide/keras/masking_and_padding)• Recurrent Neural Networks (RNN) with Keras (https://www.tensorflow.org/guide/keras/rnn)• Transfer learning and fine-tuning (https://www.tensorflow.org/guide/keras/transfer_learning)• Tokenizing with TF Text (https://www.tensorflow.org/text/guide/tokenizers)	<ul style="list-style-type: none">• Image captioning with visual attention (https://www.tensorflow.org/tutorials/text/image_captioning)• Integrated gradients (https://www.tensorflow.org/tutorials/interpretability/integrated_gradients)• Load text (https://www.tensorflow.org/tutorials/load_data/load_text)• Playing CartPole with the Actor-Critic Method (https://www.tensorflow.org/tutorials/reinforcement_learning/dl_playing_cartpole_with_the_actor_critic_method)• Word2Vec (https://www.tensorflow.org/tutorials/text/word_embeddings)

Given a tensor `input`, this operation inserts a dimension of length 1 at the dimension index `axis` of `input`'s shape. The dimension index follows Python indexing rules: It's zero-based, a negative index it is counted backward from the end.

This operation is useful to:

- Add an outer "batch" dimension to a single element.
- Align axes for broadcasting.
- To add an inner vector length axis to a tensor of scalars.

If you have a single image of shape [height, width, channels]:

```
>>> image = tf.zeros([10,10,3])
```

You can add an outer batch axis by passing `axis=0`:

```
>>> tf.expand_dims(image, axis=0).shape.as_list()
[1, 10, 10, 3]
```

The new axis location matches Python `list.insert(axis, 1)`:

```
>>> tf.expand_dims(image, axis=1).shape.as_list()
[10, 1, 10, 3]
```

Following standard Python indexing rules, a negative `axis` counts from the end so `axis=-1` adds an inner most dimension:

```
>>> tf.expand_dims(image, -1).shape.as_list()
[10, 10, 3, 1]
```

This operation requires that `axis` is a valid index for `input.shape`, following Python indexing rules:

```
-1-tf.rank(input) <= axis <= tf.rank(input)
```

This operation is related to:

- [`tf.squeeze`](https://www.tensorflow.org/api_docs/python/tf/squeeze) (https://www.tensorflow.org/api_docs/python/tf/squeeze), which removes dimensions of size 1.
- [`tf.reshape`](https://www.tensorflow.org/api_docs/python/tf/reshape) (https://www.tensorflow.org/api_docs/python/tf/reshape), which provides more flexible reshaping capability.
- [`tf.sparse.expand_dims`](https://www.tensorflow.org/api_docs/python/tf/sparse/expand_dims) (https://www.tensorflow.org/api_docs/python/tf/sparse/expand_dims), which provides this functionality for [`tf.SparseTensor`](https://www.tensorflow.org/api_docs/python/tf/sparse/SparseTensor) (https://www.tensorflow.org/api_docs/python/tf/sparse/SparseTensor)

Args

<code>input</code>	A <code>Tensor</code> .
<code>axis</code>	Integer specifying the dimension index at which to expand the shape of <code>input</code> . Given an input of D dimensions, <code>axis</code> must be in range <code>[-(D+1), D]</code> (inclusive).
<code>name</code>	Optional string. The name of the output <code>Tensor</code> .

Returns

A tensor with the same data as `input`, with an additional dimension inserted at the index specified by `axis`.

Raises

<code>ValueError</code>	If <code>axis</code> is not specified.
<code>InvalidArgumentError</code>	If <code>axis</code> is out of range $[-(D+1), D]$.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (<https://www.apache.org/licenses/LICENSE-2.0>). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (<https://developers.google.com/site-policies>). Java is a registered trademark of Oracle and/or its affiliates. Some content is licensed under the [numpy license](https://numpy.org/doc/stable/license.html) (<https://numpy.org/doc/stable/license.html>).

Last updated 2021-05-25 UTC.