TensorFlow   API r1.4

tf.boolean_mask

```
boolean_mask(
    tensor,
    mask,
    name='boolean_mask'
)
```

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

See the guide: Tensor Transformations > Slicing and Joining

Apply boolean mask to tensor. Numpy equivalent is `tensor[mask]` .

```
# 1-D example
tensor = [0, 1, 2, 3]
mask = np.array([True, False, True, False])
boolean_mask(tensor, mask)  # [0, 2]
```

In general, `0 < dim(mask) = K <= dim(tensor)` , and `mask` 's shape must match the first K dimensions of `tensor` 's shape. We then have: `boolean_mask(tensor, mask)[i, j1,...,jd] = tensor[i1,...,iK,j1,...,jd]` where `(i1,...,iK)` is the ith `True` entry of `mask` (row-major order).

Args:

- `tensor` : N-D tensor.
- `mask` : K-D boolean tensor, K <= N and K must be known statically.
- `name` : A name for this operation (optional).

Returns:

(N-K+1)-dimensional tensor populated by entries in `tensor` corresponding to `True` values in `mask` .

Raises:

- `ValueError` : If shapes do not conform.

Examples:

```
# 2-D example
tensor = [[1, 2], [3, 4], [5, 6]]
mask = np.array([True, False, True])
boolean_mask(tensor, mask)  # [[1, 2], [5, 6]]
```

Last updated November 2, 2017.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**