tf.one_hot

```
one_hot(
    indices,
    depth,
    on_value=None,
    off_value=None,
    axis=None,
    dtype=None,
    name=None
)
```

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

See the guide: Tensor Transformations > Slicing and Joining

Returns a one-hot tensor.

The locations represented by indices in `indices` take value `on_value`, while all other locations take value `off_value`.

`on_value` and `off_value` must have matching data types. If `dtype` is also provided, they must be the same data type as specified by `dtype`.

If `on_value` is not provided, it will default to the value `1` with type `dtype`

If `off_value` is not provided, it will default to the value `0` with type `dtype`

If the input `indices` is rank `N`, the output will have rank `N+1`. The new axis is created at dimension `axis` (default: the new axis is appended at the end).

If `indices` is a scalar the output shape will be a vector of length `depth`

If `indices` is a vector of length `features`, the output shape will be:

```
  features x depth if axis == -1
  depth x features if axis == 0
```

If `indices` is a matrix (batch) with shape `[batch, features]`, the output shape will be:

```
  batch x features x depth if axis == -1
  batch x depth x features if axis == 1
  depth x batch x features if axis == 0
```

If `dtype` is not provided, it will attempt to assume the data type of `on_value` or `off_value`, if one or both are passed in. If none of `on_value`, `off_value`, or `dtype` are provided, `dtype` will default to the value `tf.float32`.

> ★ **Note:** If a non-numeric data type output is desired (`tf.string`, `tf.bool`, etc.), both `on_value` and `off_value` *must* be provided to `one_hot`.

For example:

```
indices = [0, 1, 2]
depth = 3
tf.one_hot(indices, depth)  # output: [3 x 3]
# [[1., 0., 0.],
#  [0., 1., 0.],
#  [0., 0., 1.]]

indices = [0, 2, -1, 1]
depth = 3
tf.one_hot(indices, depth,
           on_value=5.0, off_value=0.0,
           axis=-1)  # output: [4 x 3]
# [[5.0, 0.0, 0.0],  # one_hot(0)
#  [0.0, 0.0, 5.0],  # one_hot(2)
#  [0.0, 0.0, 0.0],  # one_hot(-1)
#  [0.0, 5.0, 0.0]]  # one_hot(1)

indices = [[0, 2], [1, -1]]
depth = 3
tf.one_hot(indices, depth,
           on_value=1.0, off_value=0.0,
           axis=-1)  # output: [2 x 2 x 3]
# [[[1.0, 0.0, 0.0],   # one_hot(0)
#   [0.0, 0.0, 1.0]],  # one_hot(2)
#  [[0.0, 1.0, 0.0],   # one_hot(1)
#   [0.0, 0.0, 0.0]]]  # one_hot(-1)
```

Args:

- `indices`: A `Tensor` of indices.
- `depth`: A scalar defining the depth of the one hot dimension.
- `on_value`: A scalar defining the value to fill in output when `indices[j] = i`. (default: 1)
- `off_value`: A scalar defining the value to fill in output when `indices[j] != i`. (default: 0)
- `axis`: The axis to fill (default: -1, a new inner-most axis).
- `dtype`: The data type of the output tensor.

Returns:

- `output`: The one-hot tensor.

Raises:

- `TypeError`: If dtype of either `on_value` or `off_value` don't match `dtype`
- `TypeError`: If dtype of `on_value` and `off_value` don't match one another

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**