tf.nn.local_response_normalization

Contents

Aliases:

- `tf.nn.local_response_normalization`
- `tf.nn.lrn`

```
local_response_normalization(
    input,
    depth_radius=5,
    bias=1,
    alpha=1,
    beta=0.5,
    name=None
)
```

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

See the guide: Neural Network > Normalization

Local Response Normalization.

The 4-D `input` tensor is treated as a 3-D array of 1-D vectors (along the last dimension), and each vector is normalized independently. Within a given vector, each component is divided by the weighted, squared sum of inputs within `depth_radius`. In detail,

```
sqr_sum[a, b, c, d] =
    sum(input[a, b, c, d - depth_radius : d + depth_radius + 1] ** 2)
output = input / (bias + alpha * sqr_sum) ** beta
```

For details, see Krizhevsky et al., ImageNet classification with deep convolutional neural networks (NIPS 2012).

Args:

- `input`: A `Tensor`. Must be one of the following types: `float32`, `half`. 4-D.
- `depth_radius`: An optional `int`. Defaults to `5`. 0-D. Half-width of the 1-D normalization window.
- `bias`: An optional `float`. Defaults to `1`. An offset (usually positive to avoid dividing by 0).
- `alpha`: An optional `float`. Defaults to `1`. A scale factor, usually positive.
- `beta`: An optional `float`. Defaults to `0.5`. An exponent.
- `name`: A name for the operation (optional).

Returns:

A `Tensor`. Has the same type as `input`.

Last updated November 2, 2017.