O PyTorch Q Search Docs Notes [+] Language Bindings [+] Python API [-] torch torch.nn torch.nn.functional torch.Tensor Tensor Attributes Tensor Views torch.autograd torch.cuda torch.cuda.amp torch.backends torch.distributed torch.distributed.algorithms.join torch.distributed.elastic torch.distributed.fsdp torch.distributed.optim torch.distributions torch.fft

torch.futures

torch.fx

torch.hub

torch.jit

torch.linalg

torch.monitor

torch.special

torch.overrides

torch.package

torch.profiler

torch.nn.init

torch.onnx

torch.optim

Docs > torch.nn > MaxPool2d

Ecosystem

Get Started

MAXPOOL2D

return_indices=False, ceil_mode=False) [SOURCE]

CLASS torch.nn.MaxPool2d(kernel_size, stride=None, padding=0, dilation=1,

Mobile

Applies a 2D max pooling over an input signal composed of several input planes.

Blog

In the simplest case, the output value of the layer with input size (N,C,H,W), output (N,C,H_{out},W_{out}) and kernel_size (kH,kW) can be precisely described as:

Tutorials

Docs 🗸

GitHub

>_

Resources ~

$$egin{aligned} out(N_i, C_j, h, w) &= \max_{m=0,\ldots,kH-1} \max_{n=0,\ldots,kW-1} \ & ext{input}(N_i, C_j, ext{stride}[0] imes h + m, ext{stride}[1] imes w + n) \end{aligned}$$

If padding is non-zero, then the input is implicitly padded with negative infinity on both sides for padding number of points. dilation controls the spacing between the kernel points. It is harder to describe, but this link has a nice visualization of what dilation does.

• NOTE

When ceil_mode=True, sliding windows are allowed to go off-bounds if they start within the left padding or the input. Sliding windows that would start in the right padded region are ignored.

The parameters kernel_size, stride, padding, dilation can either be:

- a single int in which case the same value is used for the height and width dimension
- a tuple of two ints in which case, the first int is used for the height dimension, and the second int for the width dimension

Parameters

- **kernel_size** the size of the window to take a max over
- **stride** the stride of the window. Default value is kernel_size
- **padding** implicit zero padding to be added on both sides
- **dilation** a parameter that controls the stride of elements in the window
- return_indices if True, will return the max indices along with the outputs. Useful for torch.nn.MaxUnpool2d later
- **ceil_mode** when True, will use *ceil* instead of *floor* to compute the output shape

Shape:

- Input: (N,C,H_{in},W_{in}) or (C,H_{in},W_{in})
- ullet Output: (N,C,H_{out},W_{out}) or (C,H_{out},W_{out}) , where

$$H_{out} = \left\lfloor rac{H_{in} + 2 * ext{padding}[0] - ext{dilation}[0] imes (ext{kernel_size}[0] - 1) - 1}{ ext{stride}[0]} + 1
ight
floor$$
 $W_{out} = \left\lfloor rac{W_{in} + 2 * ext{padding}[1] - ext{dilation}[1] imes (ext{kernel_size}[1] - 1) - 1}{ ext{stride}[1]} + 1
ight
floor$

Examples:

```
>>> # pool of square window of size=3, stride=2
>>> m = nn.MaxPool2d(3, stride=2)
>>> # pool of non-square window
>>> m = nn.MaxPool2d((3, 2), stride=(2, 1))
>>> input = torch.randn(20, 16, 50, 32)
>>> output = m(input)
```

Previous Next >

© Copyright 2019, Torch Contributors.

View Docs >

Built with Sphinx using a theme provided by Read the Docs.

Docs **Tutorials** Resources

Access comprehensive developer documentation for Get in-depth tutorials for beginners and advanced PyTorch developers

View Tutorials >

Features

Find development resources and get your questions answered

View Resources >

Stay Connected PyTorch Resources Get Started Email Address **Tutorials**

Docs

Ecosystem Discuss Github Issues Blog Contributing Brand Guidelines



 \rightarrow