nn.space_to_depth

```

Copy fnspace_to_depth(tensor:@Tensor, blocksize:usize)->Tensor;

```

SpaceToDepth rearranges blocks of spatial data into depth. More specifically, this op outputs a copy of the input tensor where values from the height and width dimensions are moved to the depth dimension.

Args

- tensor
- (@Tensor
-) - The input tensor of [N,C,H,W], where N is the batch axis, C is the channel or depth, H is the height and W is the width.
- blocksize
- (usize
-) - The size of the blocks to move along [blocksize, blocksize].
-

Returns

ATensor of [N, C * blocksize * blocksize, H/blocksize, W/blocksize].

Examples

```

Copy usecore::array::{ArrayTrait,SpanTrait}; useorion::operators::tensor::{TensorTrait,Tensor}; useorion::operators::tensor::{I8Tensor,I8TensorAdd}; useorion::numbers::NumberTrait; useorion::operators::nn::NNTrait; useorion::operators::nn::I8NN; useorion::numbers::FixedTrait;

fnspace_to_depth_example()->Tensor { letmutshape=ArrayTrait:::new(); shape.append(1); shape.append(2); shape.append(2); shape.append(4);

letmutdata=ArrayTrait::new(); data.append(-3); data.append(0); data.append(0); data.append(0); data.append(-1); data.append(1); data.append(-2); data.append(-3); data.append(2); data.append(-2); data.append(-3); data.append(-3); data.append(-1); data.append(0); data.append(1); data.append(-3); lettensor=TensorTrait::new(shape.span(), data.span()); returnNNTrait::space_to_depth(@tensor,2); }

                [[[[-3,0]], [[2,-3]], [[0,0]], [[-2,-3]], [[-1,-2]], [[-1,1]], [[1,-3]], [[0,-3]]]]

```

Previous nn.depth_to_space Next Machine Learning

Last updated15 days ago