tensor.qlinear_concat

```

Copy qlinear_concat(tensors:Span>, scales:Span>, zero_points:Span>, y_scale:@Tensor, y_zero_point:@Tensor, axis:usize)->Tensor::;

```

Concatenate a list of tensors after dequantizing them with their respective scales and zero_points and returns the quantized result.

Args

- tensors
- (Span>,
-) - Array of the quantized input tensors.
- scales
- (Span>,
-) - Array of the scales of the quantized input tensors.
- zero_points
- (Span>,
-) - Arrayof the zero_points of the quantized input tensors.
- y_scale
- (@Tensor
-) - Scale for output.
- y_zero_point
- (@Tensor
-) - Zero point for output.
- axis
- (usize
-) - Axis to concat on.
-

Panics

- Panic if tensor length is not greater than 1.
- Panics if dimension is not greater than axis.
-

Type Constraints

u32 tensor, not supported. fp8x23wide tensor, not supported. fp16x16wide tensor, not supported.

Returns

A newTensor concatenated quantized tensor of the dequantized input tensors.

Example

```

Copy usecore::array::{ArrayTrait,SpanTrait};

useorion::operators::tensor::{TensorTrait,Tensor,I8Tensor,FP16x16Tensor}; useorion::numbers::
{FP16x16,FP16x16Impl,FixedTrait};

fnqlinear_concat_example()->Tensor { lettensor1=TensorTrait::< i8

    ::new( shape:array![2,2].span(), data:array![ 5, 5, 5, 5, ] .span(), ); lettensor2=TensorTrait::< i8 ::new(
    shape:array![2,2].span(), data:array![ 1, 1, 1, 1, ] .span(), );

lettensors=array![tensor1, tensor2].span();

lettensor1_scale=TensorTrait::< FP16x16

    ::new(shape:array![1].span(), data:array![FixedTrait:::new(131072,false)].span(),); lettensor2_scale=TensorTrait::
    < FP16x16 ::new(shape:array![1].span(), data:array![FixedTrait::::new(262144,false)].span(),);

letscales=array![tensor1_scale, tensor2_scale].span();
```

```
lettensor1_zero_point=TensorTrait::< FP16x16

    ::new(shape:array![1].span(), data:array![FixedTrait:::new(327680,false)].span(),);
    lettensor2_zero_point=TensorTrait::< FP16x16 ::new(shape:array![1].span(), data:array!
    [FixedTrait::::new(0,false)].span(),);

letzero_points=array![tensor1_zero_point, tensor2_zero_point].span();

lety_scale=TensorTrait::< FP16x16

    ::new(shape:array![1].span(), data:array![FixedTrait:::new(262144,false)].span(),);

lety_zero_point=TensorTrait::< FP16x16

    ::new(shape:array![1].span(), data:array![FixedTrait:::new(65536,false)].span(),);

returnTensorTrait::qlinear_concat(tensors, scales, zero_points,@y_scale,@y_zero_point,0); }

                [[1,1,1,1], [2,2,2,2]]
```

Previous tensor.qlinear_matmul Next tensor.qlinear_leakyrelu

Last updated1 month ago