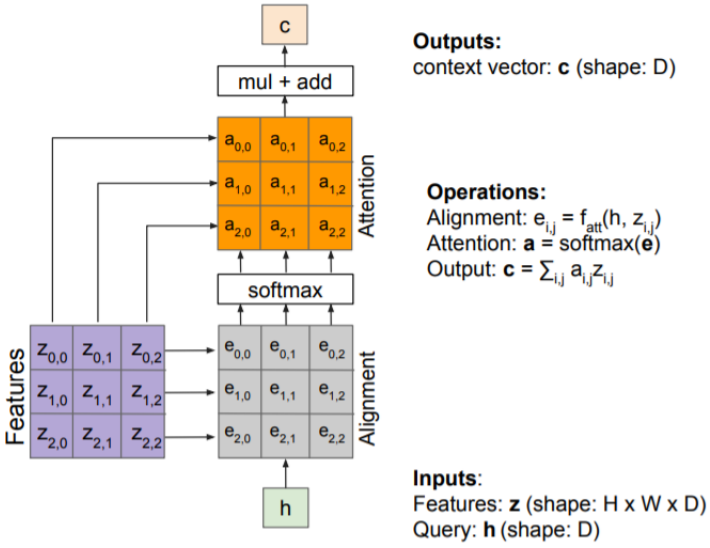
CS231n Lecture11 Attention and transformers

sgc

March 23, 2022

1 Why introduce attention

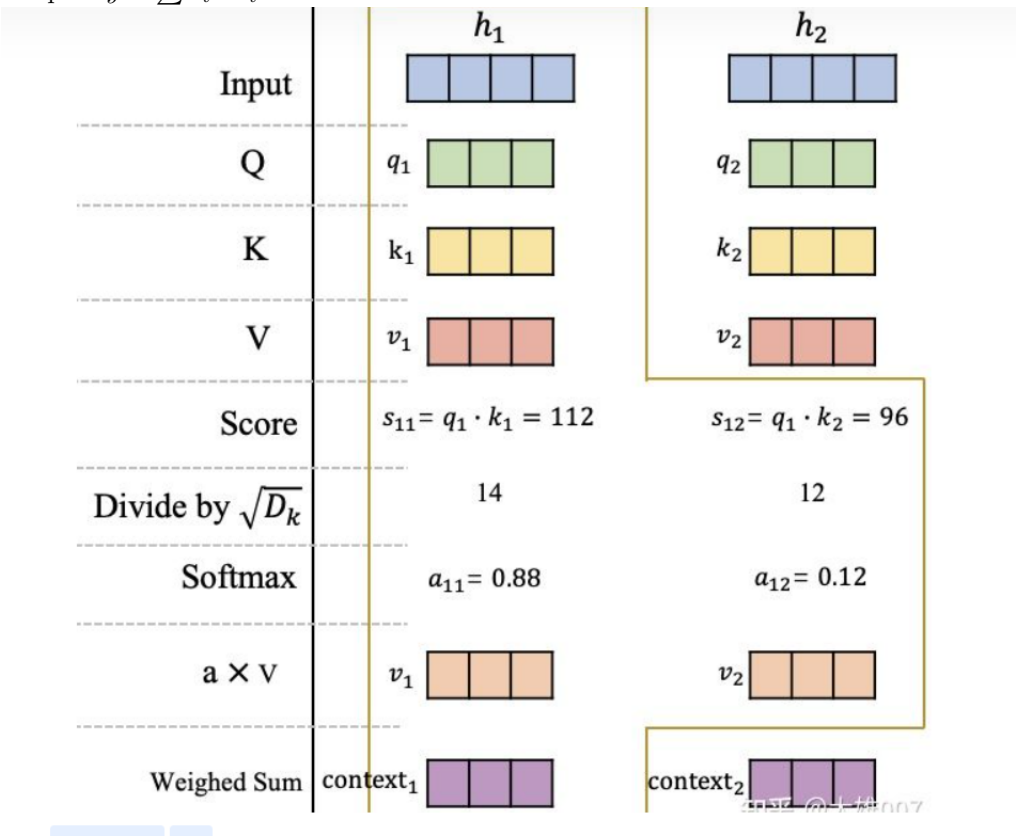
Size of context vector c is limited, Input is "bottlenecked". Attention idea: New context vector at every time step. Each c will attend to different image regions.
Compute alignments scores: $e_{t,i,j} = f_{att}(h_{t-1}, z_{i,j})$ scores of relativity in rnn $h_{t-1} is previous hidden state$
Attention:$a = softmax(e)$ regularization
context vector: $c_t = \sum a_t * z_t$
Each timestep of decoder uses a different context vector that looks at different parts of the input image.



2 Self attention layers

We can calculate the query vectors from the input vectors. Use Query-Key-Value model

Key vectors:$k = xW_k$
Value vectors:$v = xW_v$
Query vectors:$q = xW_q$
Alignment: $e_{i,j} = qj * ki/sqrt(D)$
Attention: $a = softmax(e)$
Output: $y = \sum a_t * z_t$

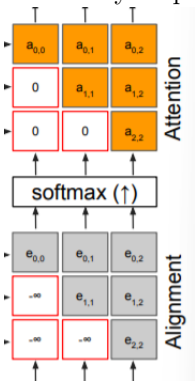


3 Position encoding

To deal with the Permutation invariant. Encode ordered sequence. Concatenate special positional encoding pj to each input vector xj.
Use a function pos: N Rd to process the position j of the vector into a d-dimensional vector.

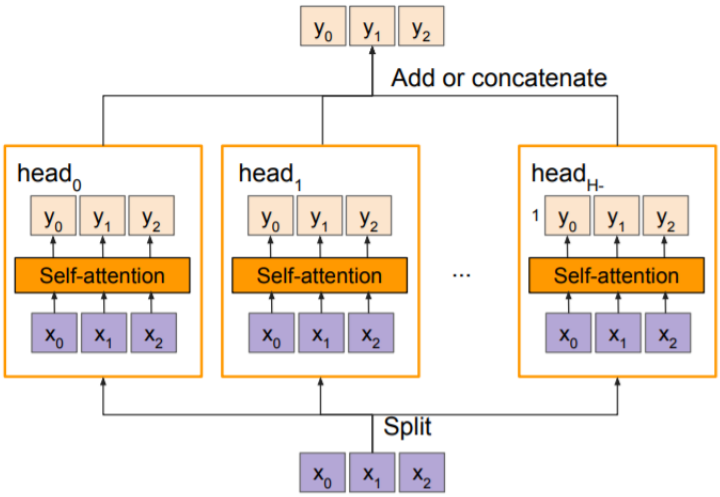
4 masked self-attention layer

Prevent vectors from looking at future vectors. - Manually set alignment scores to -infinity.Help tp deal with ordered sequence.



5 Multi-head self-attention layer

Multiple self-attention heads in parallel.

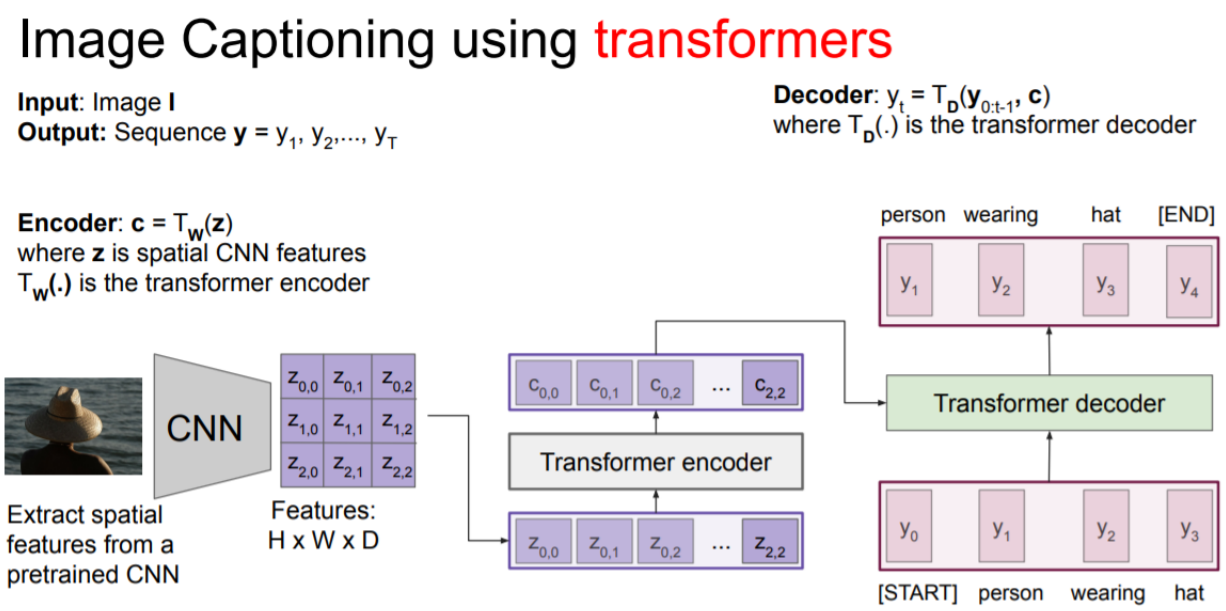


6 Advantages of transformer

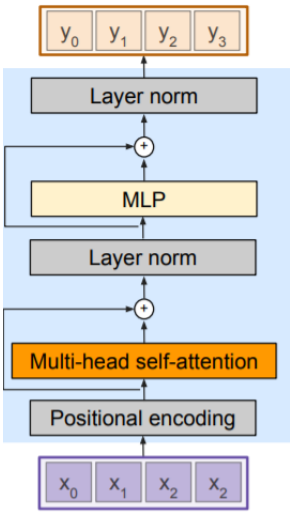
RNNs
(+) LSTMs work reasonably well for long sequences.
(-) Expects an ordered sequences of inputs
(-) Sequential computation: subsequent hidden states can only be computed after the previous ones are done.

Transformers:
(+) Good at long sequences. Each attention calculation looks at all inputs.
(+) Can operate over unordered sets or ordered sequences with positional encodings.
(+) Parallel computation: All alignment and attention scores for all inputs can be done in parallel.
(-) Requires a lot of memory: N x M alignment and attention scalers need to be calculated and stored for a single self-attention head. (but GPUs are getting bigger and better)

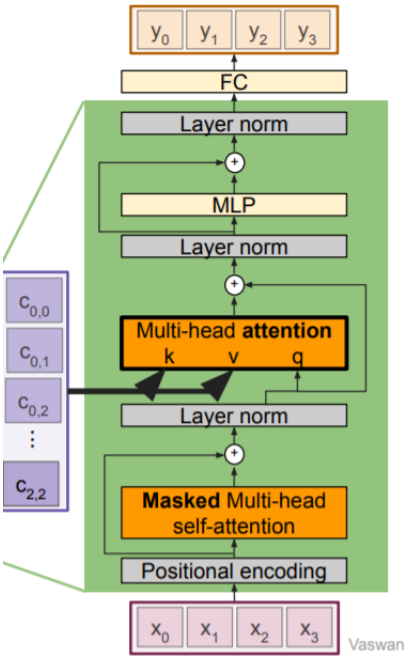
7 Transformer



The Transformer encoder block is made of N encoder blocks.
For each block:



Self-attention is the only interaction between vectors. Layer norm and MLP operate independently per vector.
The Transformer decoder block is made of N decoder blocks, too.



Multi-head attention block attends over the transformer encoder outputs(c).
And we can only use Transformer, despite of CNN.

8 Summary

Adding attention to RNNs allows them to "attend" to different parts of the input at every time step.
- The general attention layer is a new type of layer that can be used to design new neural network architectures.
- Transformers are a type of layer that uses self-attention and layer norm.
It is highly scalable and highly parallelizable.
Faster training, larger models, better performance across vision and language tasks.
They are quickly replacing RNNs, LSTMs, and may even replace convolutions.