The Transformer Architecture

Recap: Multi-Head Attention

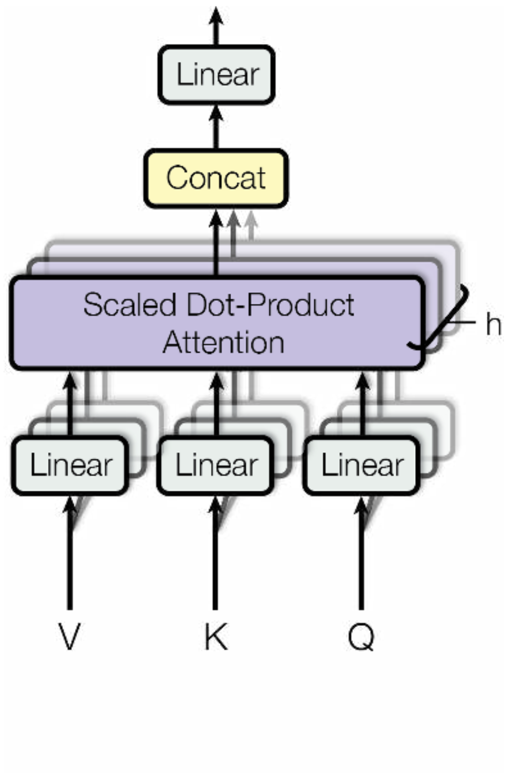
h heads, each with a set of linear projections

Additional linear projection to map to output dimension

$$\begin{bmatrix} \text{Attention}(\mathbf{XW}_{Q,1}, \mathbf{XW}_{K,1}, \mathbf{XW}_{V,1}) \\ \vdots \\ \text{Attention}(\mathbf{XW}_{Q,h}, \mathbf{XW}_{K,h}, \mathbf{XW}_{V,h}) \end{bmatrix} W_O$$

✓ Good at mixing information across multiple tokens

To represent each element in higher-dimensional space, we need to combine MHA with MLP

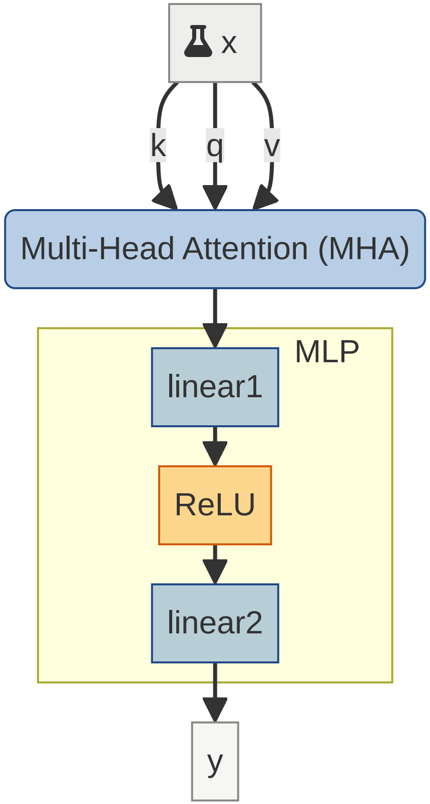


Combining MHA With MLP

Issue: vanishing gradients and activations

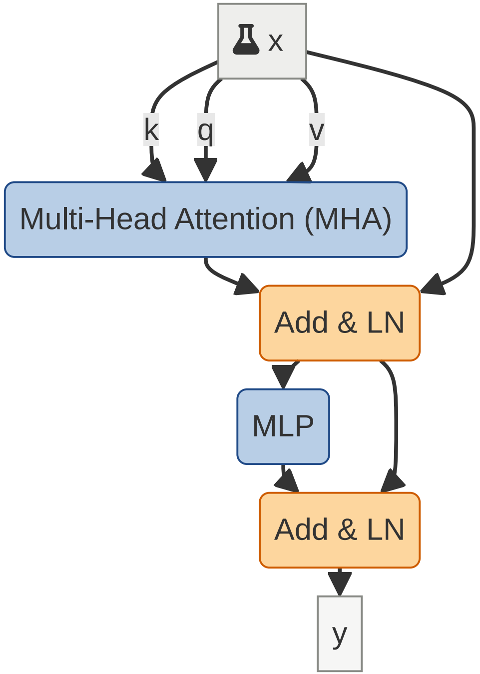
Solutions:

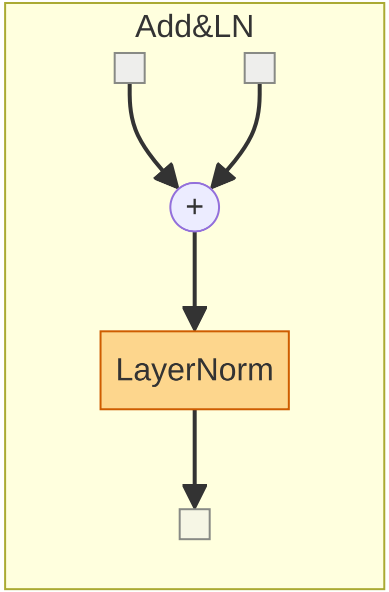
- residual connections
- normalization



Transformer Layer

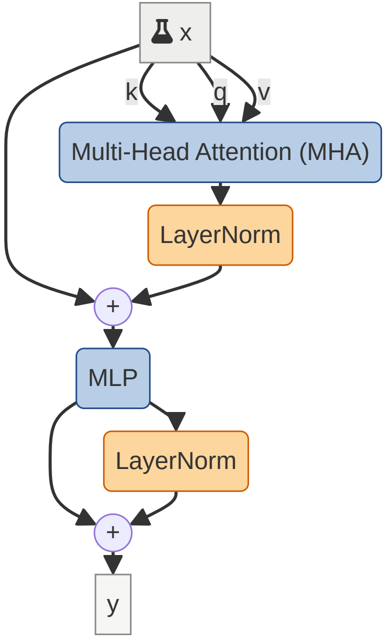
MHA + MLP + residual connection + LayerNorm



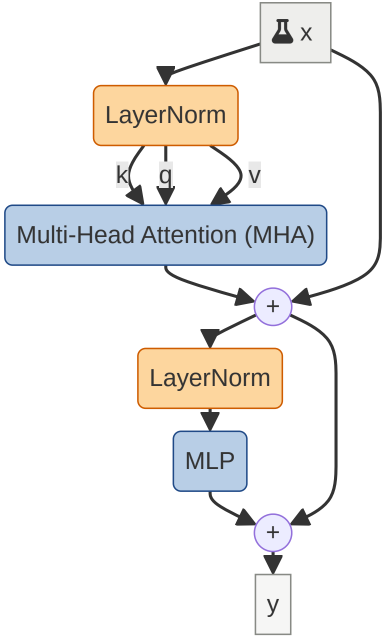


Transformer Layer: Post-Norm vs. Pre-Norm

Post-Norm (in the original Transformer[1]:)



Pre-Norm[2]:



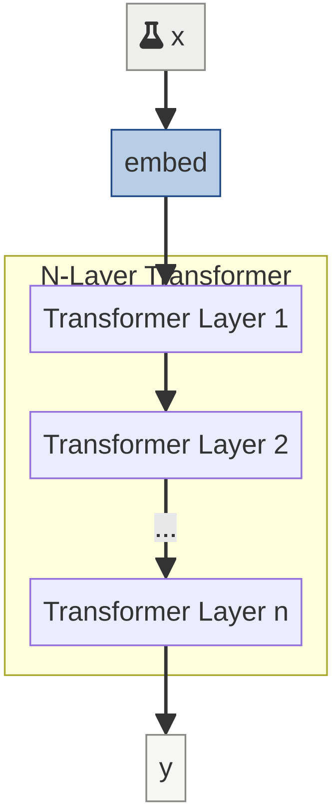
1. Vaswani, *et al.* "Attention is all you need." NeurIPS 2017 ↩️
2. Xiong, *et al.* "On layer normalization in the transformer architecture." ICML 2020 ↩️

Transformer

Inputs: a set of tokens $\{\mathbf{x}_i\}$

Outputs: another set of tokens $\{\mathbf{y}_i\}$

Simply a stack of N transformer layers

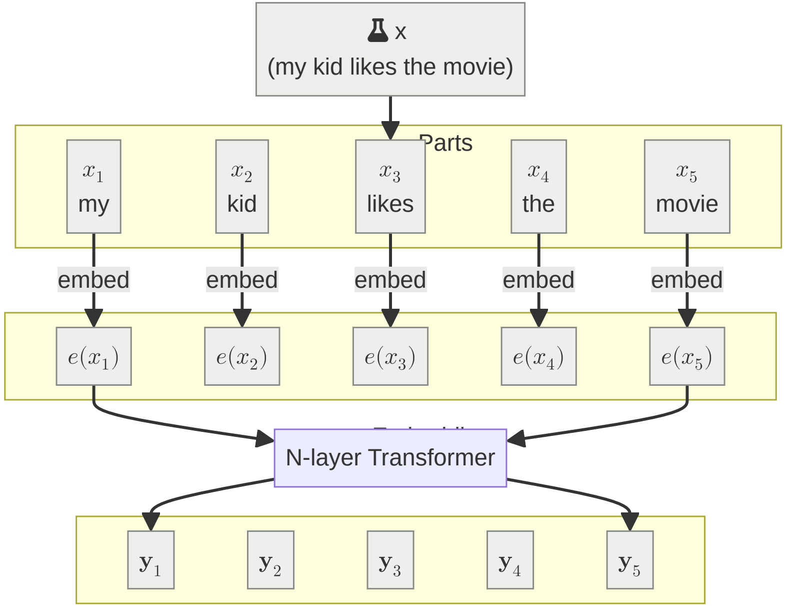


Transformer

Inputs: a set of tokens $\{\mathbf{x}_i\}$

Outputs: another set of tokens $\{\mathbf{y}_i\}$

Simply a stack of N transformer layers



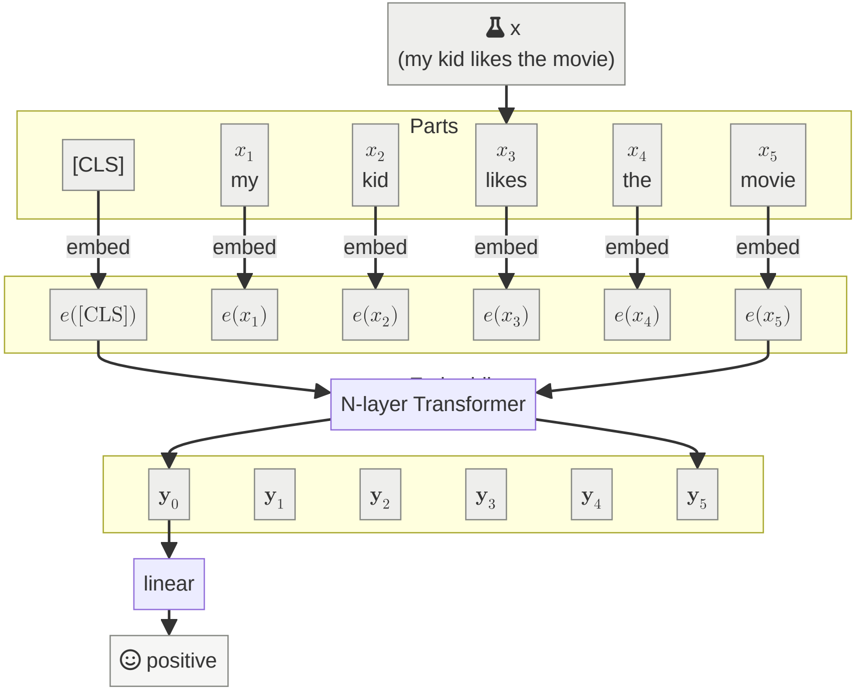
Applying Transformers to Sentiment Analysis

Examples:

🙂 `My kid likes this movie`

🙁 `My kid does not like this movie`

Prepend one more "classification" token `[CLS]`



The Transformer Architecture - TL;DR

Transformer layer = MHA + MLP + LN + residual connection

A Transformer is a stack of N transformer layers