Attention Is All You Need

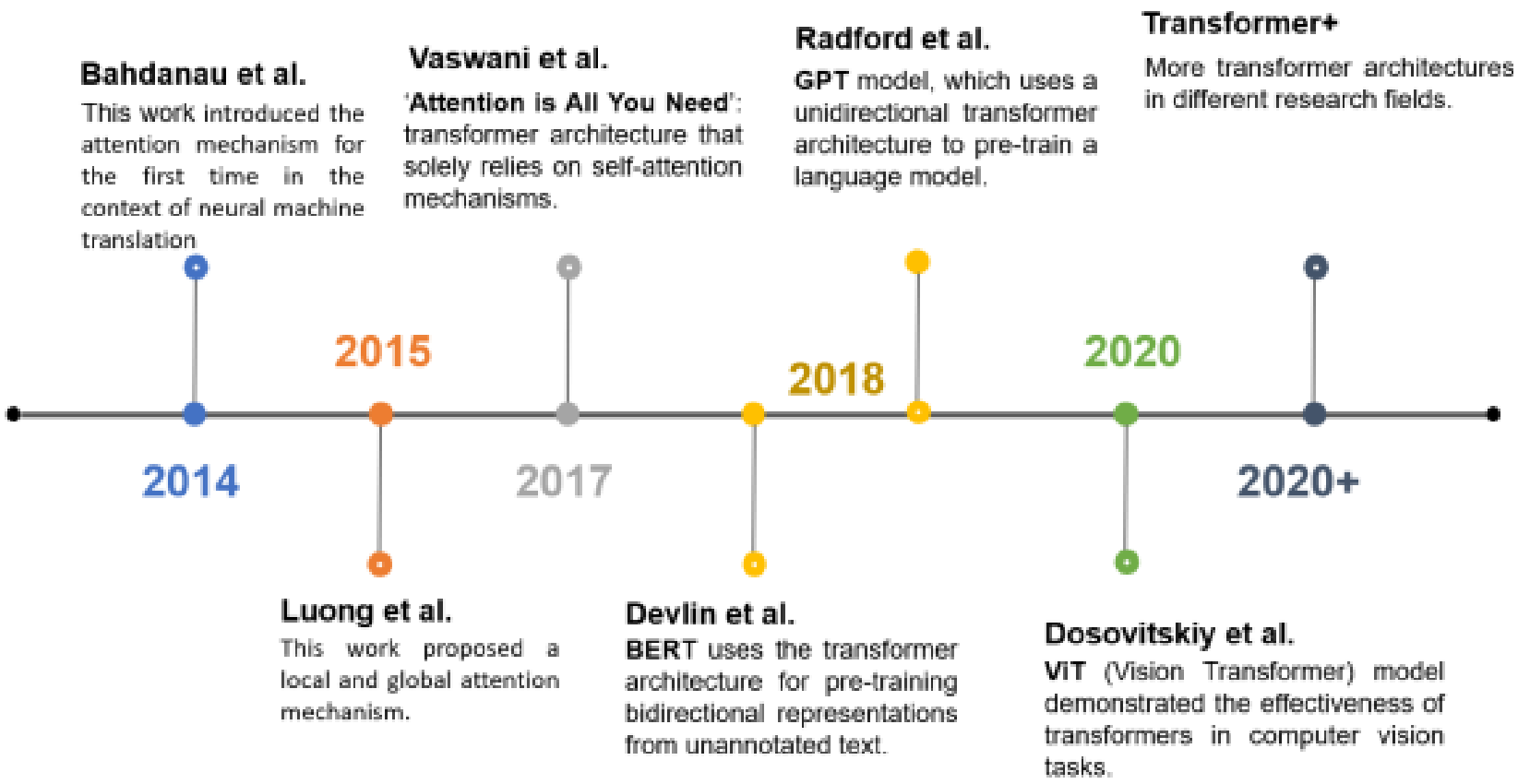
Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin
In NIPS 2017 : Advances in Neural Information Processing Systems 30

Aug 1, 2023
Sungmin Yoon

Index

- Background
- Review of Attention mechanism
- Model Architecture
- (Experiment & Result)
- Discussion

Timeline of Attention mechanism



Reference : Transformer-based models and hardware acceleration analysis in autonomous driving: A survey

Why Attention?

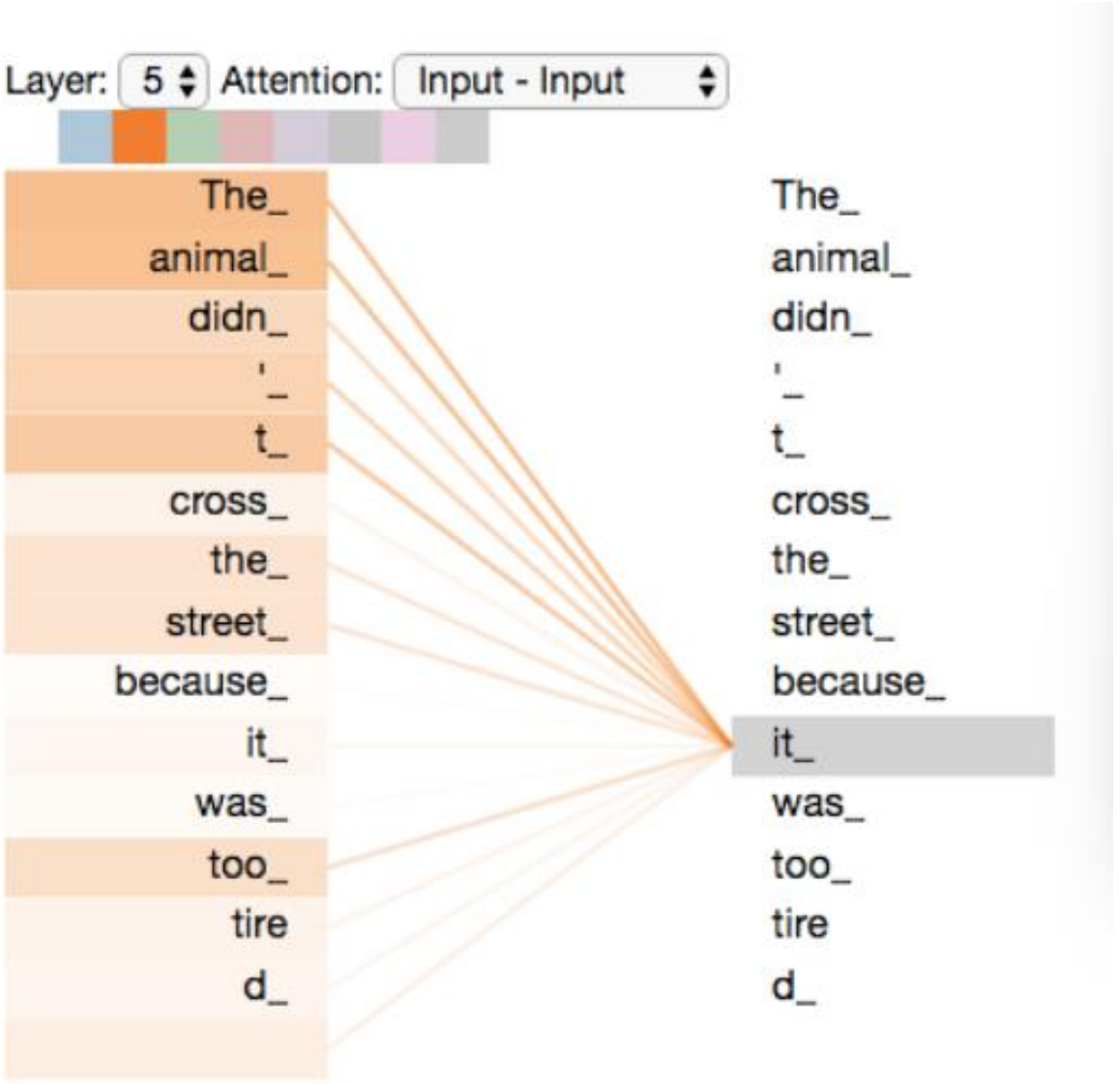
- Long-term dependency problem
  - One hidden state is not enough
- Sequential Computation
  - Transformer reduce to a constant number of operationw

Table 1: Maximum path lengths, per-layer complexity and minimum number of sequential operations for different layer types. n is the sequence length, d is the representation dimension, k is the kernel size of convolutions and r the size of the neighborhood in restricted self-attention.

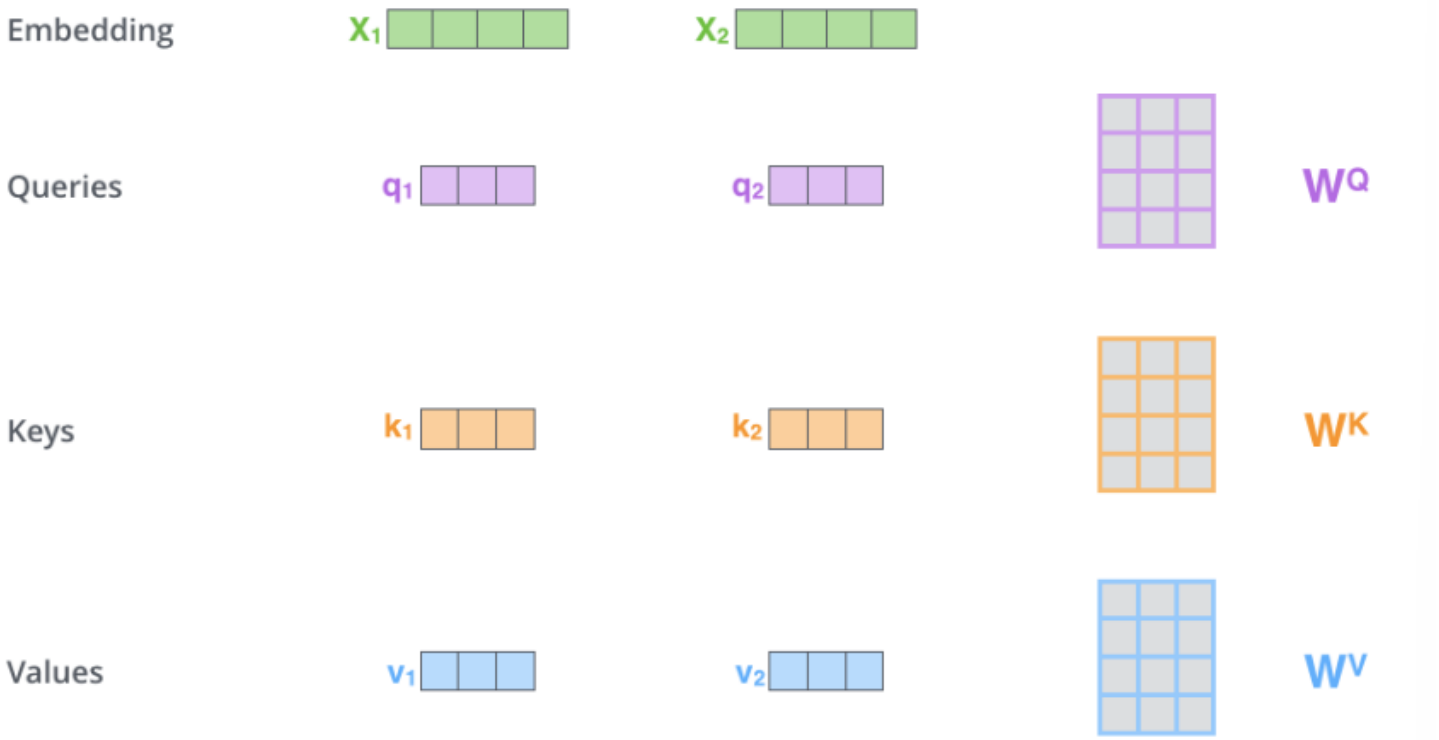
Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 \cdot d)$	$O(1)$	$O(1)$
Recurrent	$O(n \cdot d^2)$	$O(n)$	$O(n)$
Convolutional	$O(k \cdot n \cdot d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

What is Attention?

- Attention is an indicator
- We can find relevance between each word



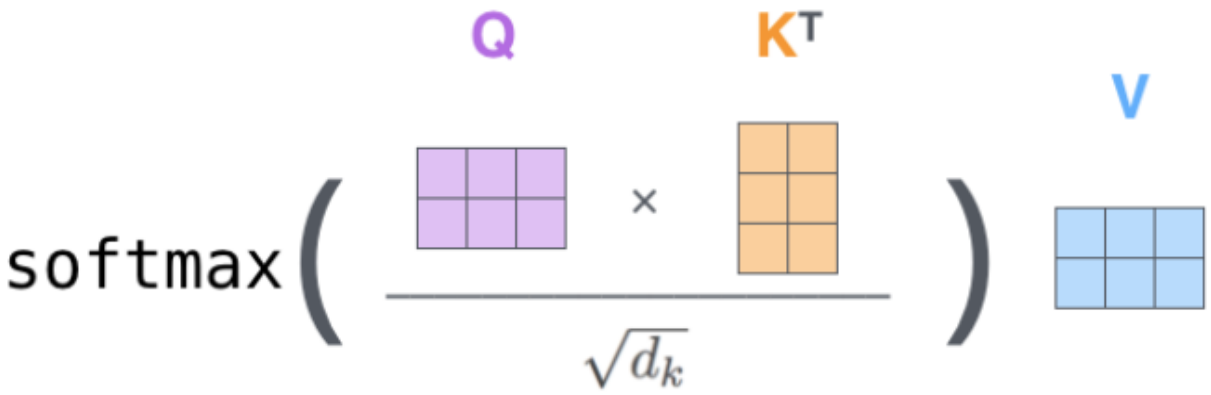
How Attention mechanism works?



1. Create three vectors Q,K,V

2. Calculate with the following formular

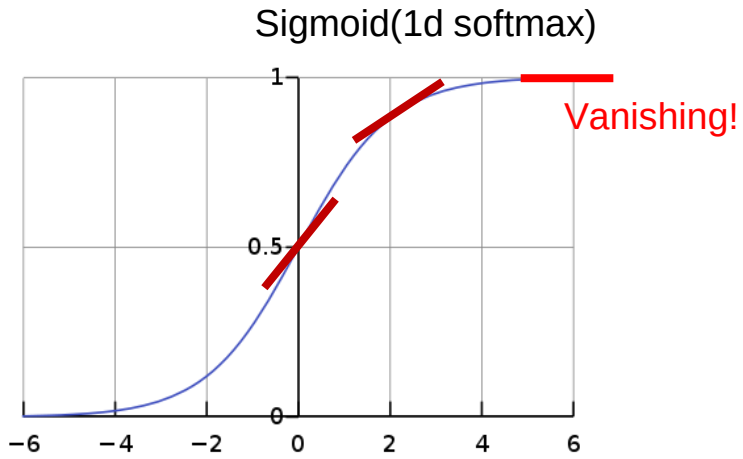
How Attention mechanism works?



The self-attention calculation in matrix form

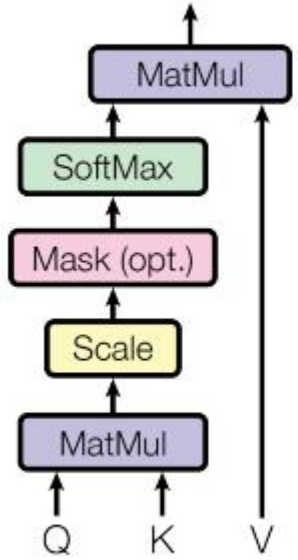
1. Create three vectors Q,K,V

2. Calculate with the following formular

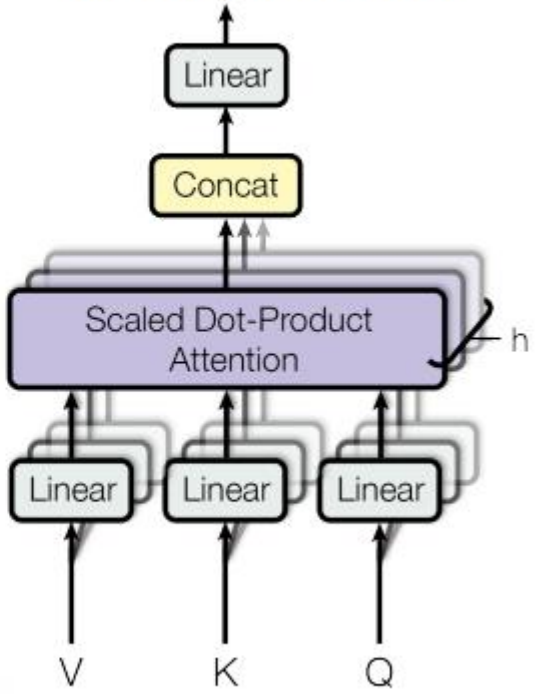


Add a Parallelism

Scaled Dot-Product Attention



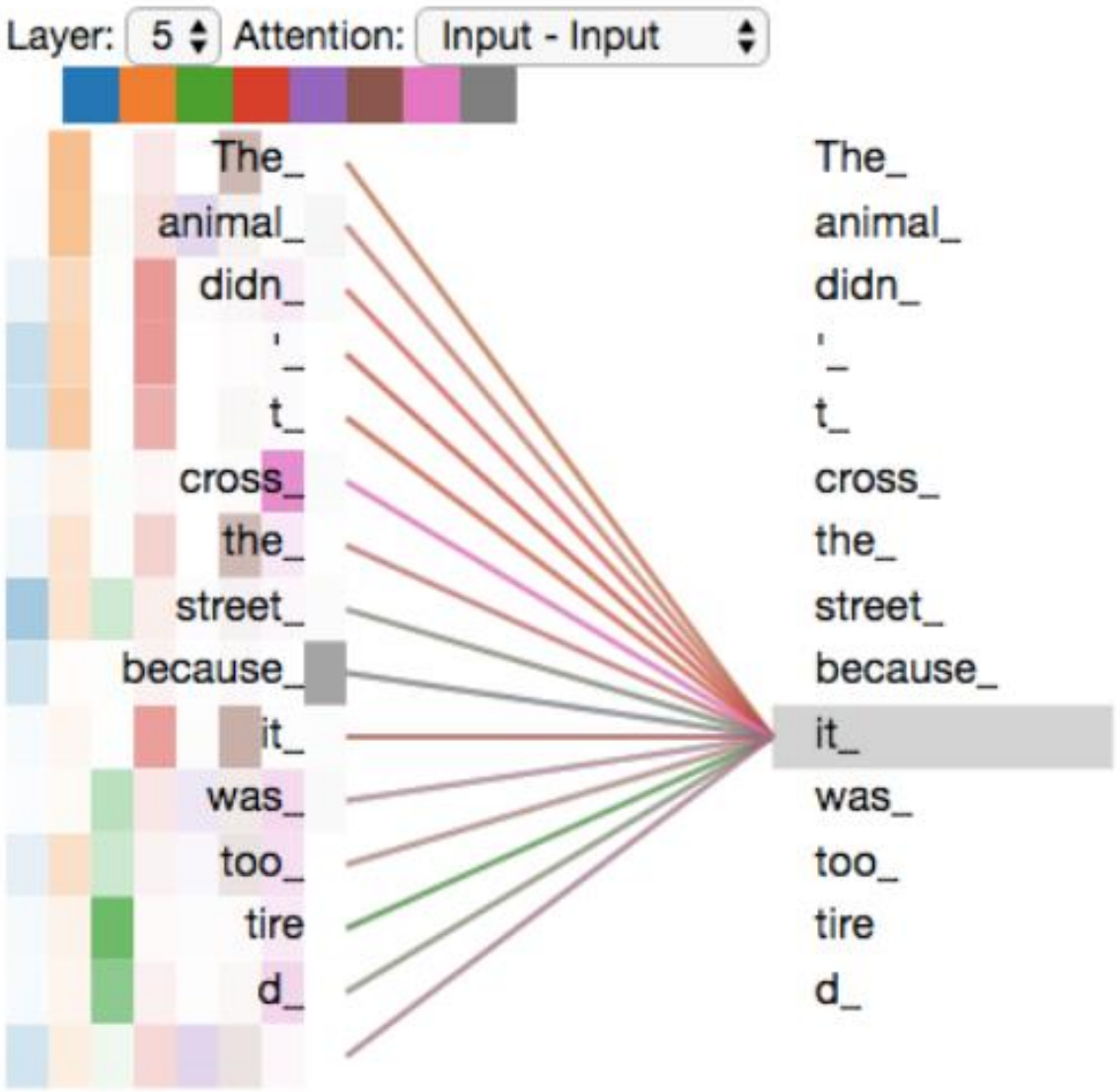
Multi-Head Attention



$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$.

Add a Parallelism



Model Architecture

- Encoder-decoder structure
- Stack size N = 6

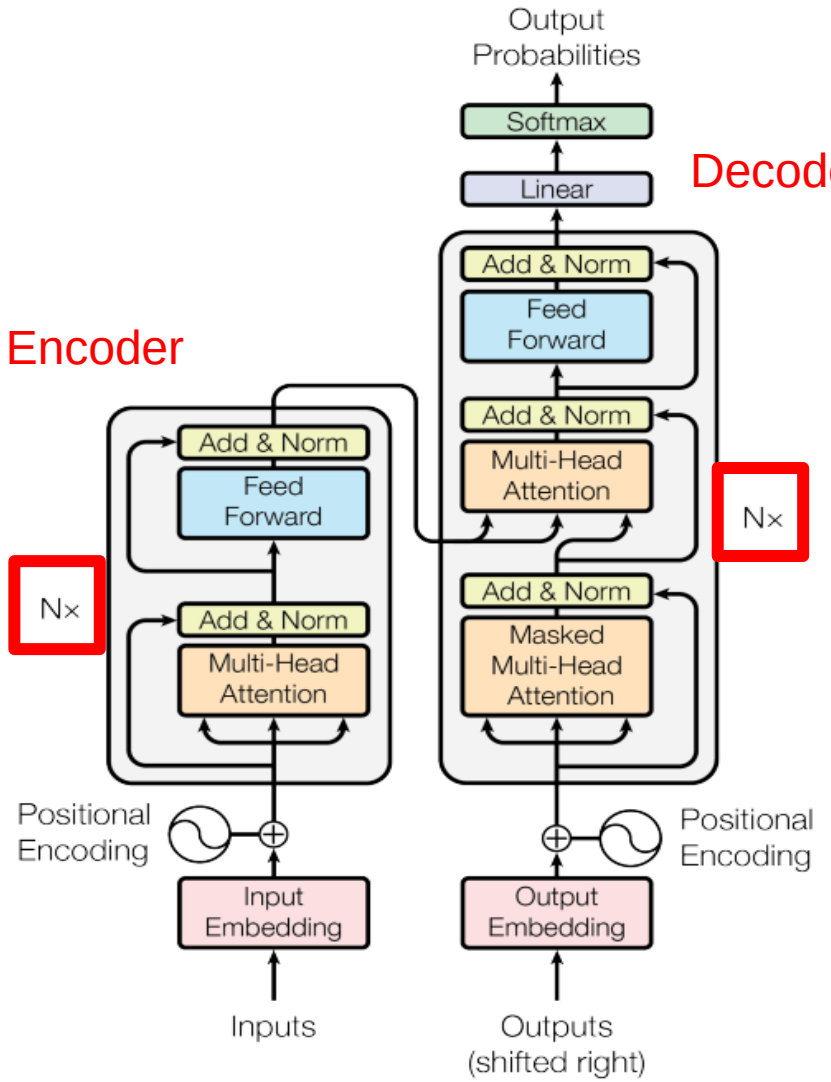


Figure 1: The Transformer - model architecture.

Model Architecture

- Encoder(2-sub layers)
  - Multi-head Self-Attention
  - Feed Forward Network(FFN) – position wise
- Residual connection
- Layer normalization

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

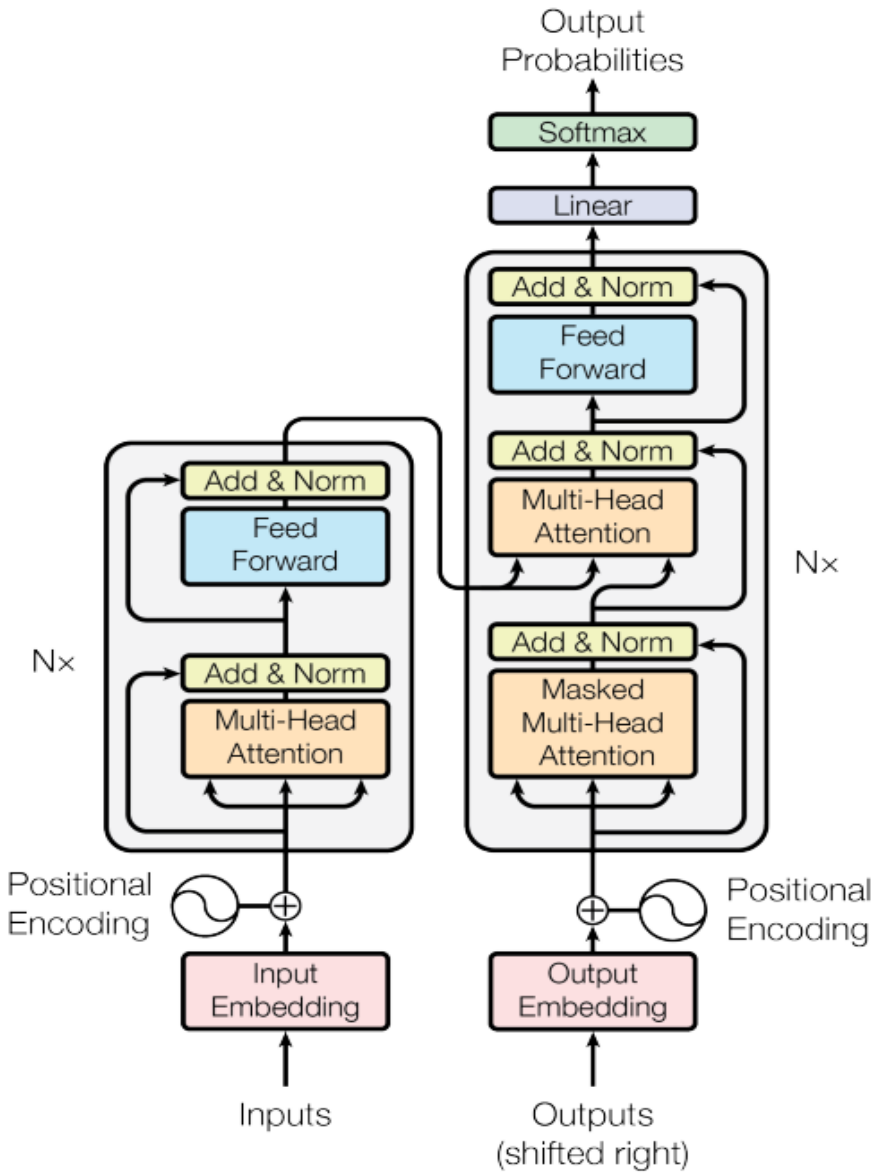


Figure 1: The Transformer - model architecture.

Model Architecture

- Decoder(3-sub layers)
  - Modified self-attention(Masking)
  - Encoder-decoder attention layer
  - FFN

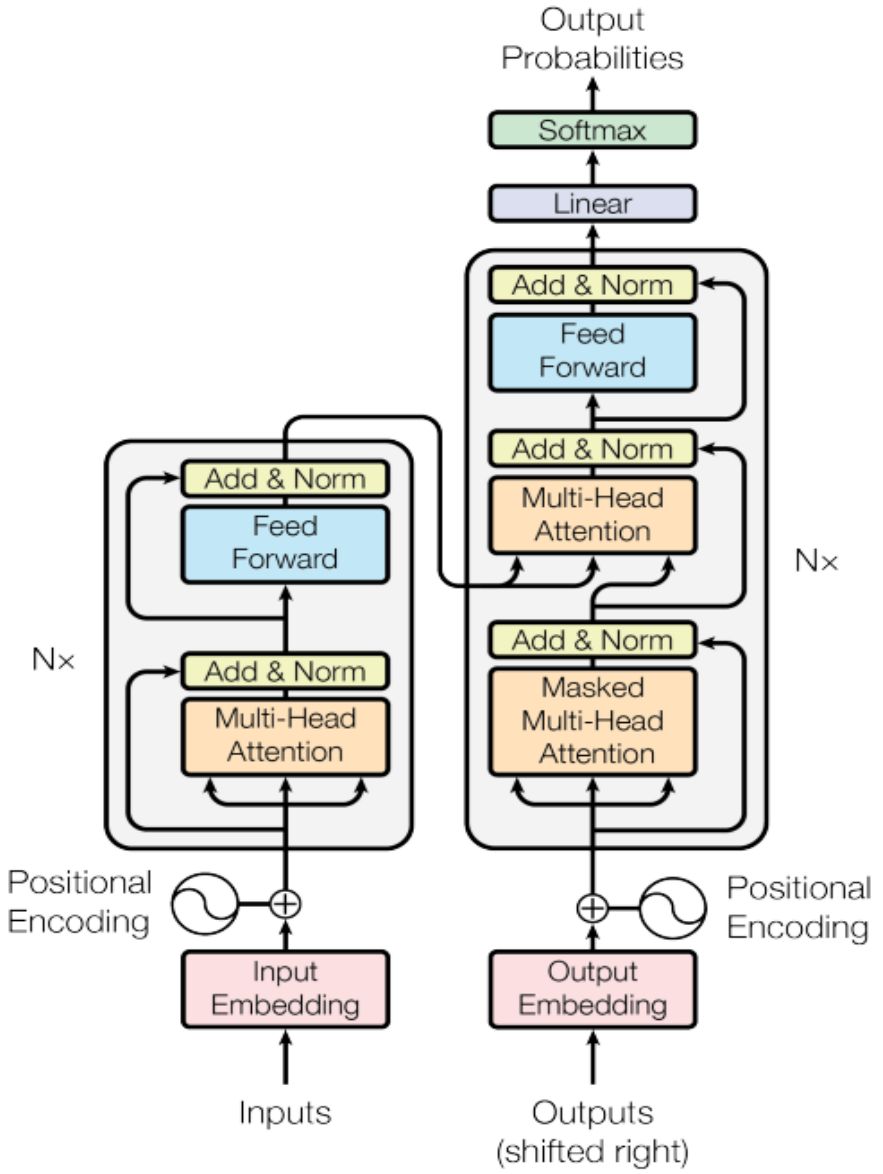


Figure 1: The Transformer - model architecture.

Self attention vs Cross attention

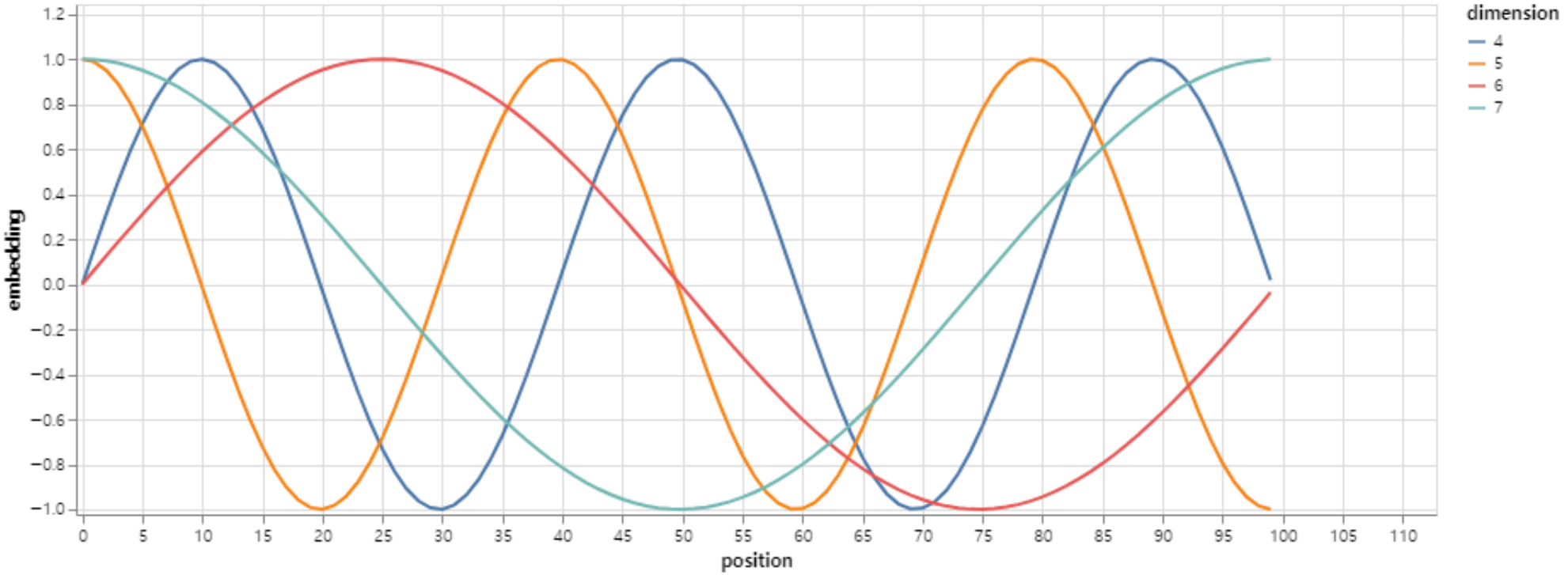
- Where the Key/Value come from?

Positional Encoding

$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

Example.
I walk my dog every day
vs
every day I walk my dog



Result with various hyper-parameters

Table 3: Variations on the Transformer architecture. Unlisted values are identical to those of the base model. All metrics are on the English-to-German translation development set, newstest2013. Listed perplexities are per-wordpiece, according to our byte-pair encoding, and should not be compared to per-word perplexities.

	N	d_{model}	d_{ff}	h	d_k	d_v	P_{drop}	ϵ_{ls}	train steps	PPL (dev)	BLEU (dev)	params $\times 10^6$
base	6	512	2048	8	64	64	0.1	0.1	100K	4.92	25.8	65
(A)				1	512	512				5.29	24.9	
				4	128	128				5.00	25.5	
				16	32	32				4.91	25.8	
				32	16	16				5.01	25.4	
(B)					16					5.16	25.1	58
					32					5.01	25.4	60
(C)	2									6.11	23.7	36
	4									5.19	25.3	50
	8									4.88	25.5	80
		256			32	32				5.75	24.5	28
		1024			128	128				4.66	26.0	168
			1024							5.12	25.4	53
			4096							4.75	26.2	90
(D)							0.0			5.77	24.6	
							0.2			4.95	25.5	
								0.0		4.67	25.3	
								0.2		5.47	25.7	
(E)	positional embedding instead of sinusoids									4.92	25.7	
big	6	1024	4096	16			0.3		300K	**4.33**	**26.4**	213

Discussion

- Good
  - Detailed and qualified result
  - Transferable -> more brain-like
  - Explanation
- Bad
  - Memory cost

Q&A

- Restricted Self Attention?
- Structure image
- Naming of Query, key, value
- Positional encoding -> what is each sinusoid and dimmension