Attention Is All You Need

Transformer

陳宜蓁 111705007、范士朋 314511051、陳大福 314511073、翁祐宸 314611104
曾耀陞 A141074、戴廷勳 A141083、謝尚哲 A141163、蔡忻恩 A141199

Outline

- Introduction
- Model Structure
- Attention
- Other modules
- Self-Attention
- Training and Results
- Conclusion

Introduction

Background:

- Neural machine translation (NMT) systems rely on RNN/CNN encoder–decoder architectures
- **Attention** mechanisms help modeling dependencies, but **only as an additional component**

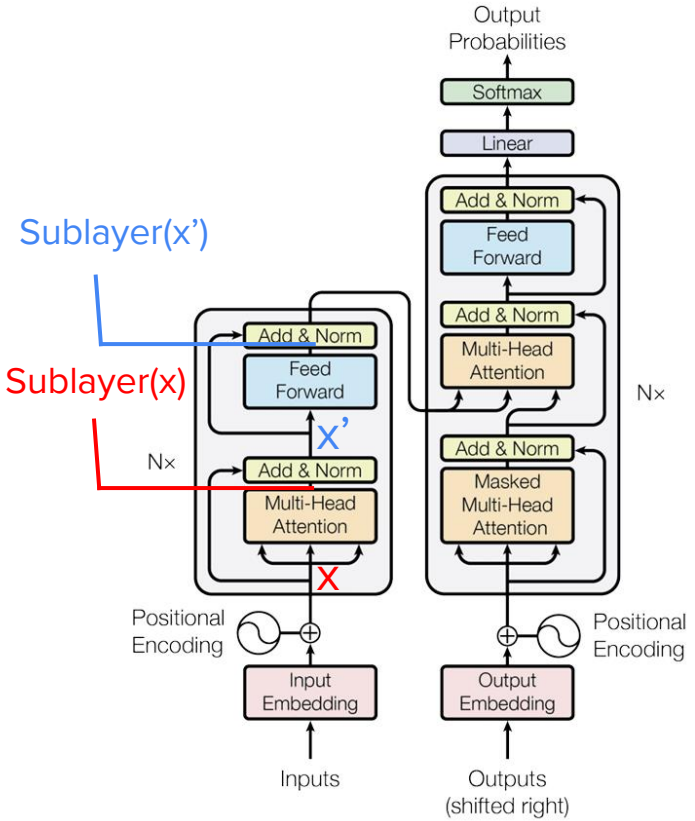
Limitation:

- Sequential computation in RNNs/CNNs **restricts parallelization and slows training**, especially for long sequences.

Solution - Transformer

- eliminates recurrence and convolutions
- **self-attention**
- constant path length
- more **parallelization**

Model Structure



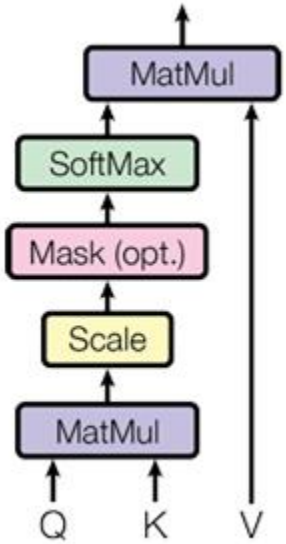
Encoder:

- a stack of N = 6 identical layers
- two sublayers: **multi-head attention**, **position-wise feed-forward network**
- residual connection followed by layer normalization
- output of each sub-layer : $\mathrm{LayerNorm}(x + \mathrm{Sublayer}(x))$

Decoder:

- a stack of N = 6 identical layers
- three sublayers: **masked multi-head self-attention,** multi-head attention, position-wise feed-forward network
- ensures output token at position i only depends on tokens before i

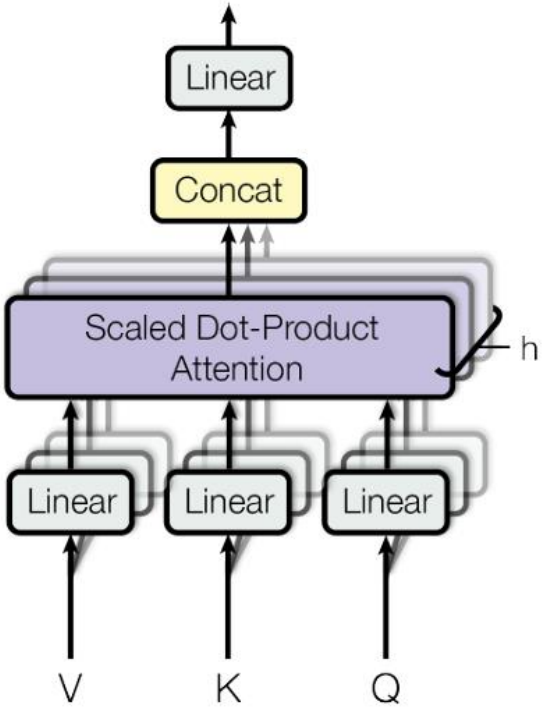
Scaled Dot-Product Attention



$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

- much faster and more space-efficient in practice than additive attention
- To prevent large dot products: $\frac{1}{\sqrt{d_k}}$

Multi-Head Attention



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

- Linearly project the queries, keys and values h times
- Allow the model to jointly attend to information from different representation subspaces at different positions.

Position-wise Feed-Forward Networks

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

- Applied to each position independently and identically
- Two linear layers with ReLU activation

  （512 → 2048 → 512）
- Enhances non-linearity and transformation capacity
- Works together with Attention to form each Transformer layer

Embeddings and Softmax

- Convert tokens into d_model-dimensional embeddings
- Shared weight matrix for input embedding, output embedding, and pre-softmax linear layer
- Embedding weights are scaled by √d_model for stable training

Positional Encoding

- Transformer needs position information (no RNN/CNN)
- Add sinusoidal positional encodings to embeddings
- Enables the model to learn relative and absolute positions

Why Self-Attention?

	RNN	CNN	Self-Attention
Path Length	O(n)	O(log n)	O(1)
Parallelism	Low	Medium	High
Long Dependency	Weak	Medium	Strong

- Enables modeling long-range dependencies in **one step**
- More parallelizable than RNN
- Constant path length between any two positions
- Handles global context efficiently

Results-Machine Translation

Training Setup

- **Dataset:**
  WMT 2014 EN-DE EN-FR
- **Optimizer:**
  Adam
- **Learning Rate:**
  Warmup + Inverse Sqrt Decay

SOTA Results

- **EN-DE Translation:**
  28.4 BLEU
- **EN-FR Translation:**
  41.0 BLEU
- **Key Conclusion:**
  Achieved a new SOTA at a fraction of the training cost.

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [15]	23.75			
Deep-Att + PosUnk [32]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [31]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [8]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [26]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [32]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [31]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [8]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.0**	$2.3 \cdot 10^{19}$	

Results-Model Variations

Attention Heads

A single head performs worse, but too many heads also degrades quality. 8 heads was optimal.

Key Dimension

Reducing the key dimension significantly hurts model quality, suggesting compatibility is non-trivial.

Model Size

Bigger models perform better. Dropout is essential to prevent overfitting.

Positional Encoding

Using Sinusoidal or Learned positional encodings resulted in nearly identical performance.

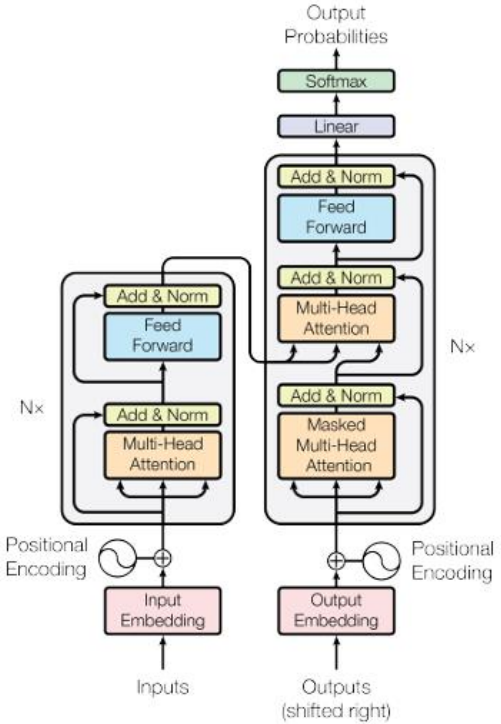
Conclusion & Impact

Contributions

- **New Architecture:**
  Based solely on attention, dropping RNNs/CNNs for parallelization.
- **Long-Range:**
  O(1) path length effectively captures long-range dependencies.
- **SOTA:**
  Faster to train and more accurate, becoming an NLP milestone.

Impact

- The foundation for subsequent models: BERT, GPT, T5, ViT.
- The core insight: **"Attention Is All You Need."**



THANKS