Attention is All You Need Analysis

Raghav Mishra

June 28, 2025

1 PART 1

1.1 First Pass Summary

Proposal of a new single network architecture, Transformer, based solely on attention mechanisms. Transformers don't rely on RNN or CNN; they use self-attention instead.

Introduction to terms like encoder-decoder structure, encoder-decoder stacks, attention function, types of attention functions, and their applications.

Comparison of self-attention layers with RNN and CNN.

The model was trained on the WMT 2014 English-German dataset containing 4.5 million sentence pairs. The WMT 2014 English-French dataset, containing 36 million sentences, was also used.

Eight NVIDIA P100 GPUs were used for model training.

The big Transformer model outperformed the best previously reported models on the WMT 2014 English-to-German translation task by more than 2.0 BLEU, achieving a new state-of-the-art score of 28.4 BLEU. On the WMT 2014 English-to-French translation task, the big model achieved a BLEU score of 41.0.

1.2 Key Architecture Components

1.3 Self Attention Mechanism

Self-attention helps the model focus on different words in a sentence when trying to understand each word.

How It Works (Step-by-Step):

1. Each word in the sentence is turned into a number vector (a list of numbers that represent meaning).

2. For every word, the model creates:

   - A query: what the word is looking for.
   - A key: what the word offers to others.
   - A value: the actual meaning/info the word gives.

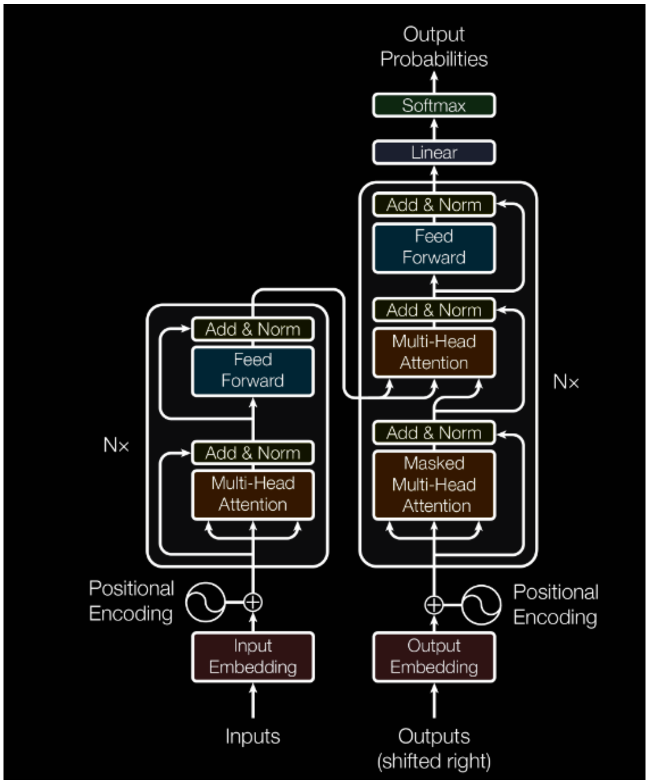

Figure 1: Transformer model architecture.

3. For a word like “sat”, the model:
   - Looks at the queries and keys to see which other words are important.
   - Calculates scores for each word in the sentence (how much attention to pay).
   - Uses those scores to make a weighted average of all the value vectors (this creates a new, richer representation of the word ”sat”).
4. This is done for every word at the same time.

1.4 Training Methodology

Trained on the standard WMT 2014 English-German dataset consisting of about 4.5 million sentence pairs. Sentences were encoded using byte-pair encoding, which has a shared source-target vocabulary of about 37,000 tokens. For English-French, the significantly larger WMT 2014 English-French dataset consisting of 36M sentences was used and split tokens into a 32,000 word-piece vocabulary. Sentence pairs were batched together by approximate sequence length. Each training batch contained a set of sentence pairs containing approximately 25,000 source tokens and 25,000 target tokens.

1.5 Result Analysis

On the WMT 2014 English to German task the model outperformed previously existing models by more than 2 BLEU score and set a new state-of-the-art BLEU score of 28.4. On the WMT 2014 English to French task the model achieved a BLEU score of 41.0.

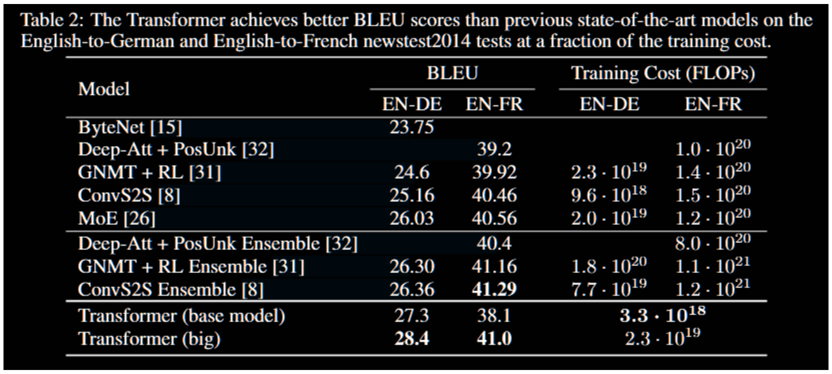

Table 2: The Transformer achieves better BLEU scores than previous state-of-the-art models on the English-to-German and English-to-French newstest2014 tests at a fraction of the training cost.

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	$\mathbf{3.3 \cdot 10^{18}}$	
Transformer (big)	**28.4**	**41.0**	$2.3 \cdot 10^{19}$	

Figure 2: Summary of Results

2 Part 2

2.1 Comparison with RNN/LSTM

RNN: Simple architecture, suitable for basic time-series data.

LSTM: Mitigates vanishing gradient problem, effective for sequence data.

Transformers: Parallelizable, captures global context, excellent scalability.

2.2 Innovation

Replaces recurrence with self-attention, enabling parallel computation.

Since Transformers have no recurrence or convolution, they use positional encoding to inject sequence order information.

Multiple attention layers run in parallel.

Encoder: Processes the input sequence and encodes it into a set of hidden representations.

Decoder: Generates the output sequence one token at a time using the encoder's output and previously generated tokens.

2.3 Modern Application

- **Natural Language Processing (NLP):** Transformers power models like GPT and BERT, enabling advanced text generation, translation, and sentiment analysis.
- **Computer Vision:** Vision Transformers (ViTs) are used for image classification and object detection, competing with convolutional neural networks (CNNs).
- **Speech Recognition:** Transformers enhance automatic speech recognition (ASR) systems, improving accuracy in voice assistants and transcription services.

2.4 Future Directions

The below diagram can be helpful in the improvements of the model.

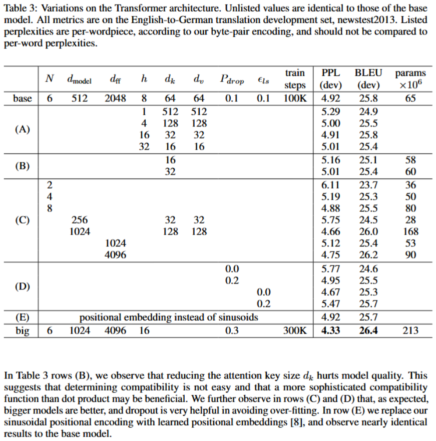

Table 3: Variations on the Transformer architecture. Unlisted values are identical to those of the base model. All metrics are on the English-to-German translation development set, newstest2013. Listed perplexities are per-wordpiece, according to our byte-pair encoding, and should not be compared to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

	N	d_{model}	d_{ff}	h	d_k	d_v	P_{drop}	ϵ_{ls}	train	PPL	BLEU	params
									steps	(dev)	(dev)	$\times10^6$
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

In Table 3 rows (B), we observe that reducing the attention key size d_k hurts model quality. This suggests that determining compatibility is not easy and that a more sophisticated compatibility function than dot product may be beneficial. We further observe in rows (C) and (D) that, as expected, bigger models are better, and dropout is very helpful in avoiding over-fitting. In row (E) we replace our sinusoidal positional encoding with learned positional embeddings [8], and observe nearly identical results to the base model.

Figure 3: Additional Results or Analysis

3 Full Analysis

3.1 RNN (Recurrent Neural Network)

- Uses recurrent connections for memory retention.
- Processes input sequences one step at a time.
- Handles short-term dependencies effectively.
- Learns via backpropagation through time (BPTT).

- Generates a sequence of hidden states for input tokens.

Recurrent models typically factor computation along the positions of symbols in the input and output sequences, which limits parallelization and makes learning long-range dependencies difficult.

3.2 LSTM (Long Short-Term Memory)

- An improved variant of RNN designed to handle long-term dependencies.
- Incorporates gates (input, forget, output) to control the flow of information.
- Better suited for tasks involving long context, such as translation and speech recognition.

3.3 GRNN (Gated Recurrent Neural Network)

- A general term that includes models like LSTM and GRU (Gated Recurrent Unit).
- Sometimes used to refer to Radial Basis Function (RBF) networks with a recurrent structure.
- Adds gating mechanisms to standard RNNs to improve performance on sequential tasks.

3.4 Sequence Transduction

Sequence transduction refers to any process that takes an input sequence and produces an output sequence. Common applications include:

- Machine translation (e.g., English to German).
- Speech-to-text systems.
- Summarization and text generation.

Traditional models like RNNs or LSTMs perform sequence transduction using sequential processing, which can be slow and hard to train over long sequences.

3.5 Transduction Modules in Transformers

In the Transformer architecture, transduction modules are the components that enable sequence-to-sequence conversion, namely:

- **Encoder:** Processes the entire input sequence in parallel using self-attention, and encodes it into a contextual representation.

- **Decoder:** Generates the output sequence, attending both to previous outputs and the encoder's representation.

These modules replace recurrence with attention mechanisms, allowing for faster training and better performance on long-range dependencies.

3.6 Related Architectures and Concepts

- **Extended Neural GPU:** Advanced neural network architecture built on neural GPU capabilities for parallel computing, algorithm learning, and memory efficiency.
- **ByteNet:** Sequence modeling architecture using convolutional neural networks for efficient sequence transduction.
- **ConvS2S:** Convolutional Sequence to Sequence models offer a middle ground between RNN-based models and attention-based Transformers, combining faster training times with effective context modeling. They integrate attention mechanisms to align input and output sequences effectively.
- **Self-Attention:** Allows a neural network to weigh the importance of different elements in an input sequence while making predictions.
- **Intra-Attention:** Attention between different positions within a single sequence.
- **Recurrent Attention Mechanism:** Combines attention with RNN mechanisms.
- **Seq Transduction Model:** Used to transform an input to output sequence.
- **Residual Connection:** Introduced by ResNet, skip connections in deep learning neural networks.
- **Decoder:** Performs multi-head attention over the output of the encoder stack.
- **Layer Normalization:** A technique used to stabilize and accelerate the training of deep neural networks. It normalizes the inputs across the features of a single data sample, rather than across a batch (as in batch normalization).
- **Attention:** Can be described as mapping a query and a set of key-value pairs to an output.
- **Multi-Head Attention:** Core component of the Transformer architecture, allowing the model to attend to different positions in a sequence simultaneously and capture diverse relationships between tokens.

- **BLEU score:** a metric used to evaluate the quality of machine-generated text—most commonly in machine translation—by comparing it to one or more reference translations written by humans.
- **Regularization:**Regularization is a technique used in machine learning to prevent overfitting, which happens when a model learns the training data too well—including noise and outliers—resulting in poor generalization to unseen data.
- **Residual Dropout:** It helps prevent overfitting by introducing randomness into the residual connections.
- **Label Smoothing:** Label Smoothing is a regularization technique used in classification tasks to make the model less confident about its predictions, thereby improving generalization.

3.7