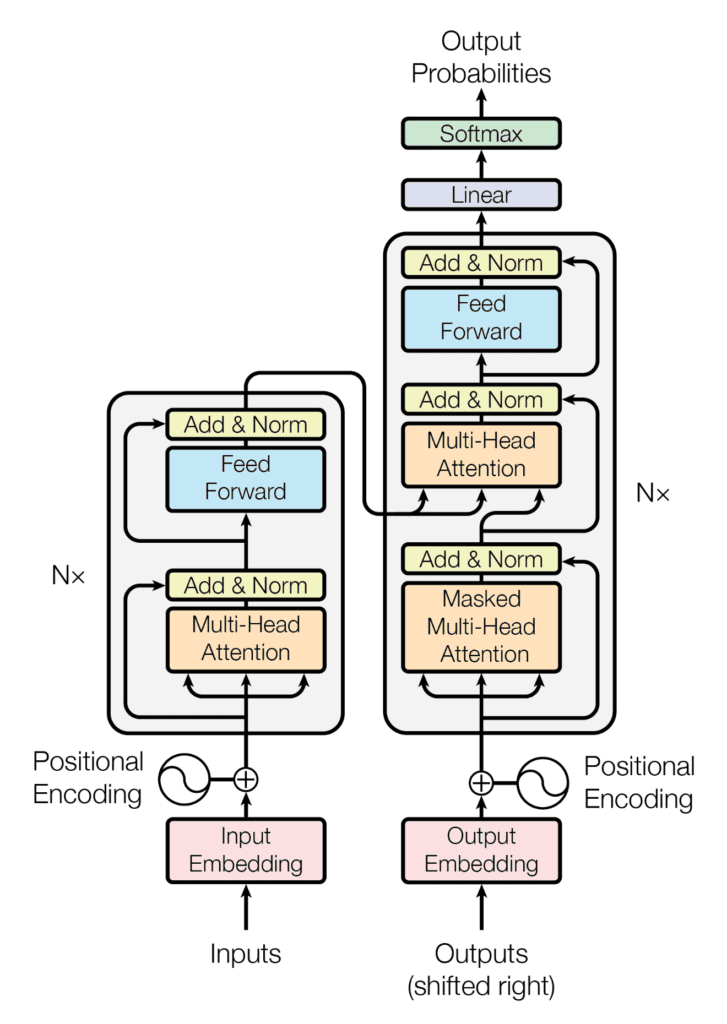
Attention Is All You Need – Summary

INTRODUCTION

This paper proposes a new network architecture called Transformer, which is based solely on attention mechanisms. When this model was compared with its recurrent and convolution counterparts, it was seen that:

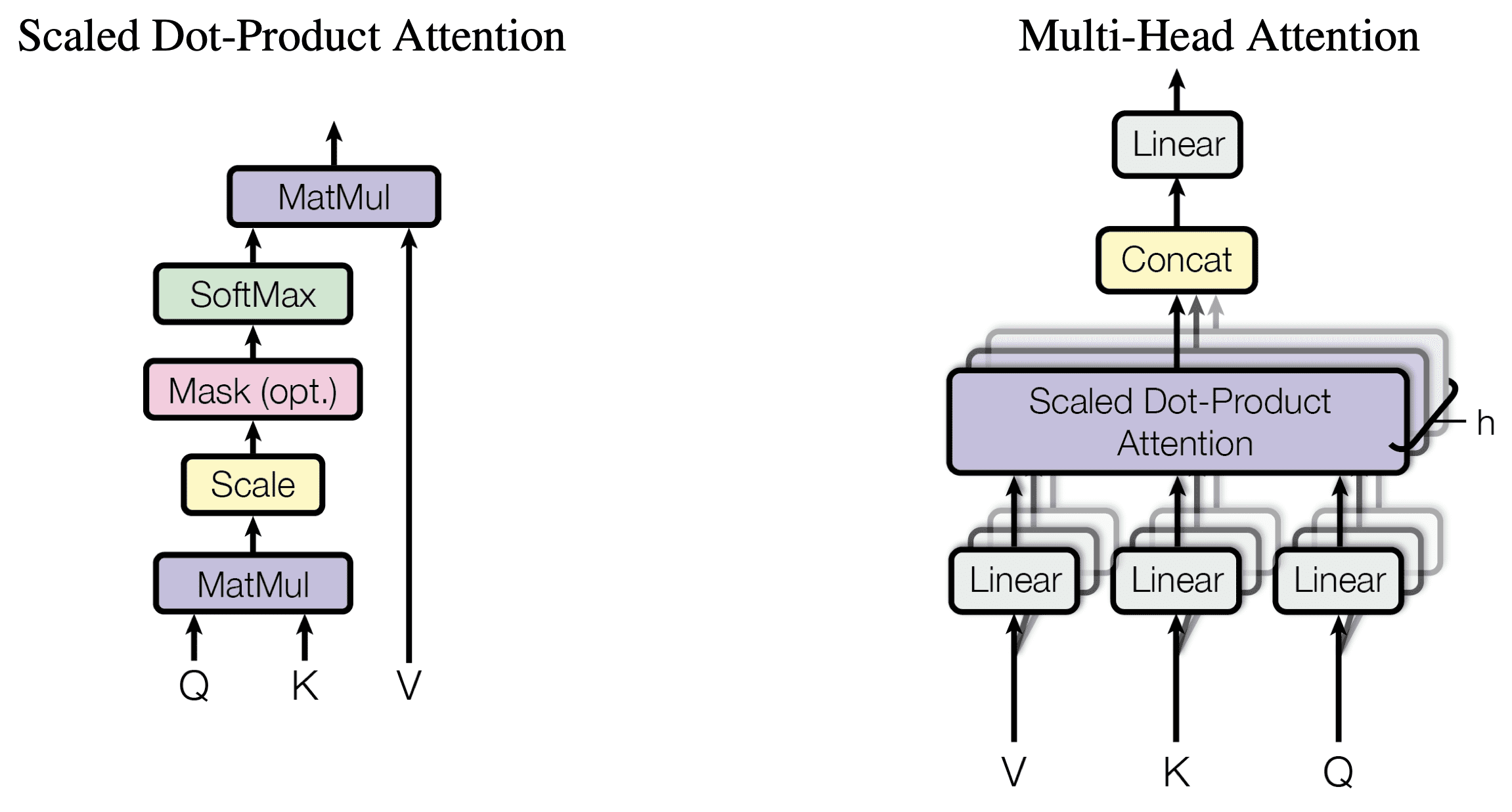
* This model is superior in quality while being more parallelizable and requiring significantly less time to train.
* It generalizes well on other tasks using both large and limited training data.

This architecture relies heavily on the concept of self-attention, relating different positions of a single sequence in order to compute a representation of the sequence. This allows the model to effectively capture long-range dependencies and contextual information.

ARCHITECTURE

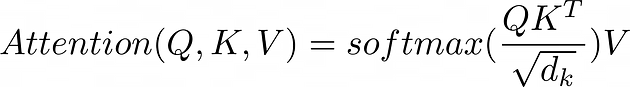
The transformer model consists of an encoder-decoder structure. Both the encoder and decoder are composed of a stack of N=6 identical layers, each containing a multi-head self-attention mechanism and pointwise, fully connected layers feed-forward network.

Additionally, the decoder contains a third layer that performs multi-head attention over the output of the encoder layer. During training, the self-attention sub-layer is modified to prevent positions from attending to subsequent positions. This masking ensures that the prediction of particular position are dependent only on the known predictions of the positions before it.



**Attention**

An attention function can be described as mapping a query and a set of key-value pairs to an output, where they are all vectors. The output is computed as a weighted sum of the values. The attention used in this paper is called “Scaled-Dot Product Attention” whose matrix of outputs is computed as such:

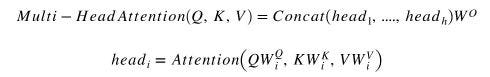


where Q, K, V denote the matrix for query, keys and values respectively and dk is the dimension of keys.

It is suspected that for large values of dk, the dot products grow large in magnitude, pushing the softmax function into regions where it has extremely small gradients. Hence, to counteract this effect, we scale the dot products by .

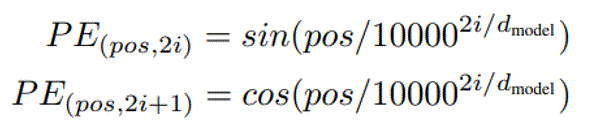
**Multi-Head attention**

Instead of performing a single attention function, it was found more beneficial to perform it h different times with different learned linear projection of the input at different positions. These output values are then concatenated and once again projected resulting in the final values.



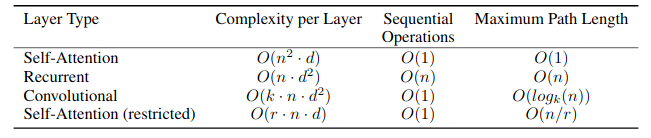
**Positional Embedding**

Since this model contains no recurrence and no convolution, positional embeddings were added in order for the model to make use of the order of the sequence. In this work, sine and cosine functions of different frequencies were chosen:

  
This function was chosen because it was hypothesized that it would allow the model to easily learn to attend by relative positions, since for any fixed offset k, PEpos+k can be represented as a linear function of PEpos . Experiments with learned embeddings were also done but the two versions produced almost identical results.

Advantage of Self-Attention when compared with recurrent and convolution layers:

* The total computational complexity per layer was less.
* The amount of computation that can be parallelized was less.
* It was able to learn long-range dependencies in the network easily. This was a challenge in RNNs and CNNs as RNNs struggle to capture long-range dependencies due to vanishing gradients and CNN has a limited receptive field window and hence, it may struggle to capture dependencies that span beyond it.



This entire model was trained on the standard WMT 2014 English-German dataset consisting of about 4.5 million sentence pairs. Adam optimizer was used with β1=0.9, β2=0.98 and ε=10-9. The learning rate was varied according to the formula:

