Introduction to Transformer

Sanmay Ganguly
(sanmay@iitk.ac.in)

Machine Learning for Particle and Astroparticle Physics, IOPB 2024

04/07/2024

One of the most celebrated paper of modern times.

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

*Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

†Work performed while at Google Brain.

‡Work performed while at Google Research.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Attention is all you need

A Vaswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, AN Gomez, Ł Kaiser, I Polosukhin

Advances in neural information processing systems, 2017 • proceedings.neurips.cc

Abstract

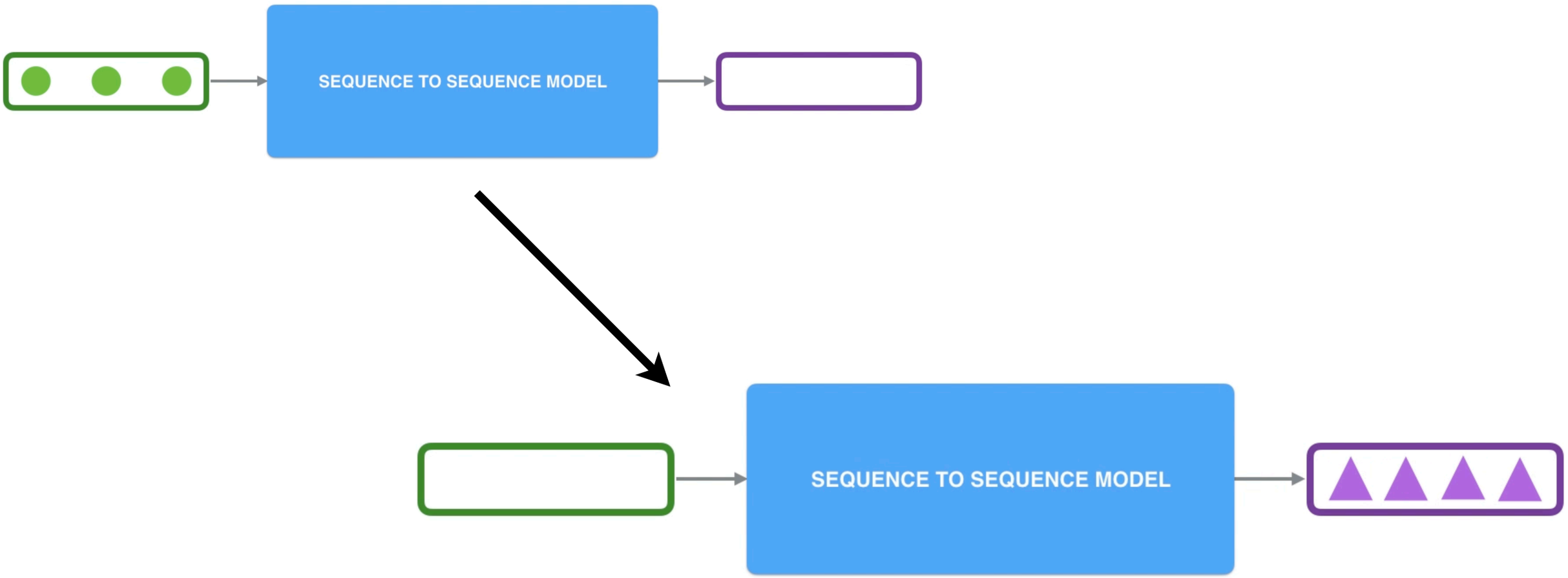
The dominant sequence transduction models are based on complex recurrent orconvolutional neural networks in an encoder and decoder configuration. The best performing such models also connect the encoder and decoder through an attentionm echanisms. We propose a novel, simple network architecture based solely onan attention mechanism, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superiorin quality while being more

SHOW MORE

☆ Save Cite Cited by 124014 Related articles All 91 versions »

Sequence to Sequence mapping

A sequence is an ordered set, can be words, letters, images, ... any data-structure .. but the positional order has to have a right human interpretable meaning.

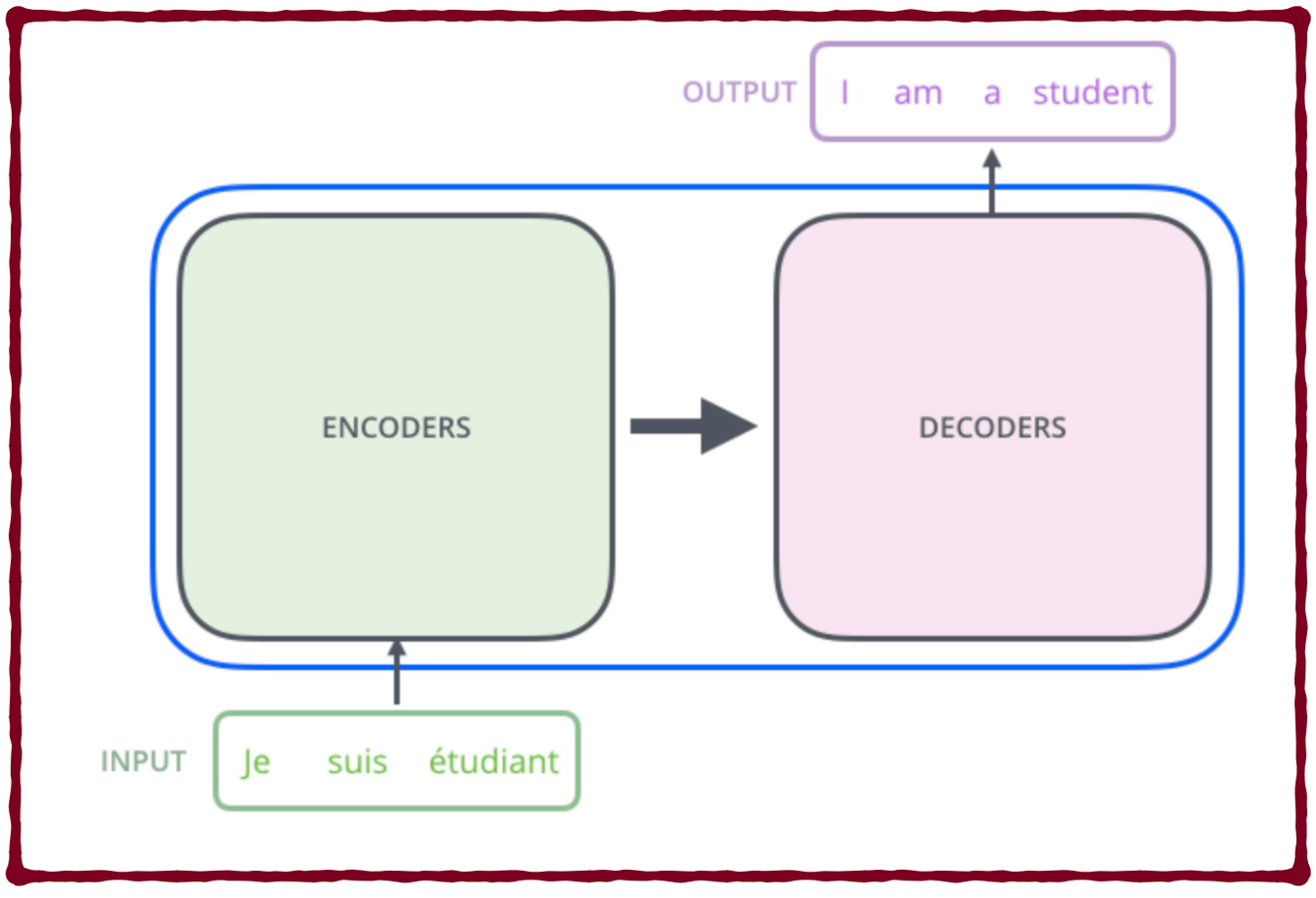


For a neural machine translation, in NLP context this sequence is a sentence with the right ordering of words.

Let’s construct a model to solve it

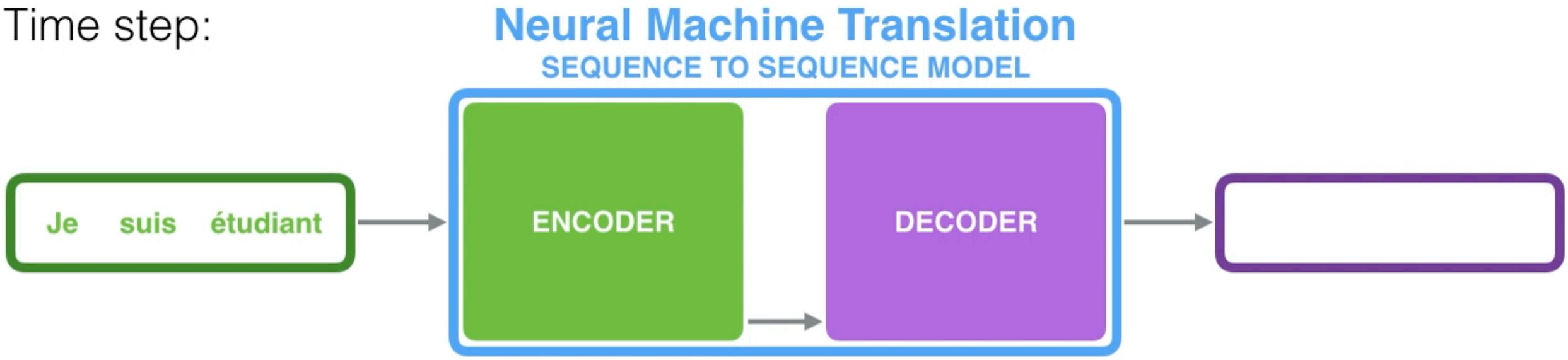
What are the general principles to code the transformer ?

https://jalammar.github.io/illustrated-transformer/

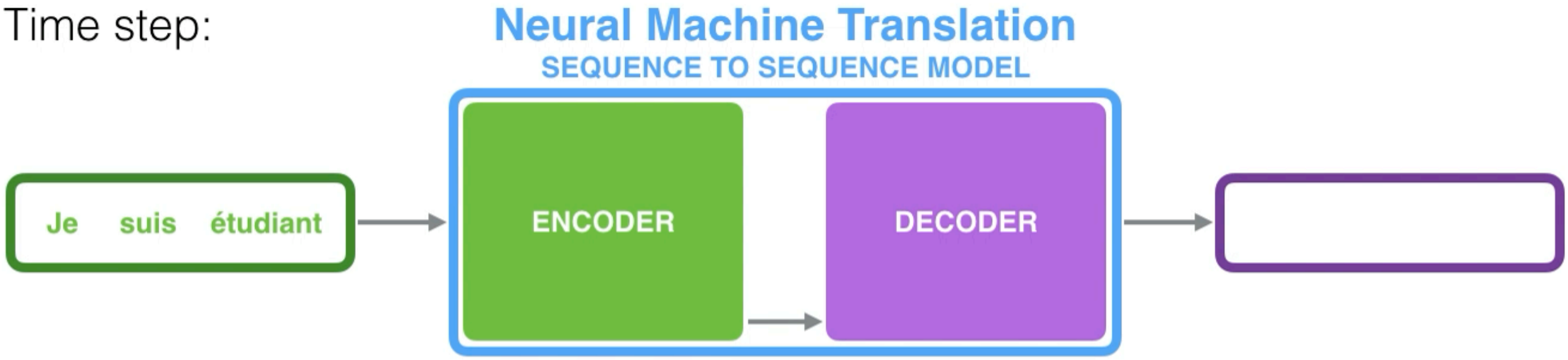


An encoder - decoder turned out to be serving this purpose best.

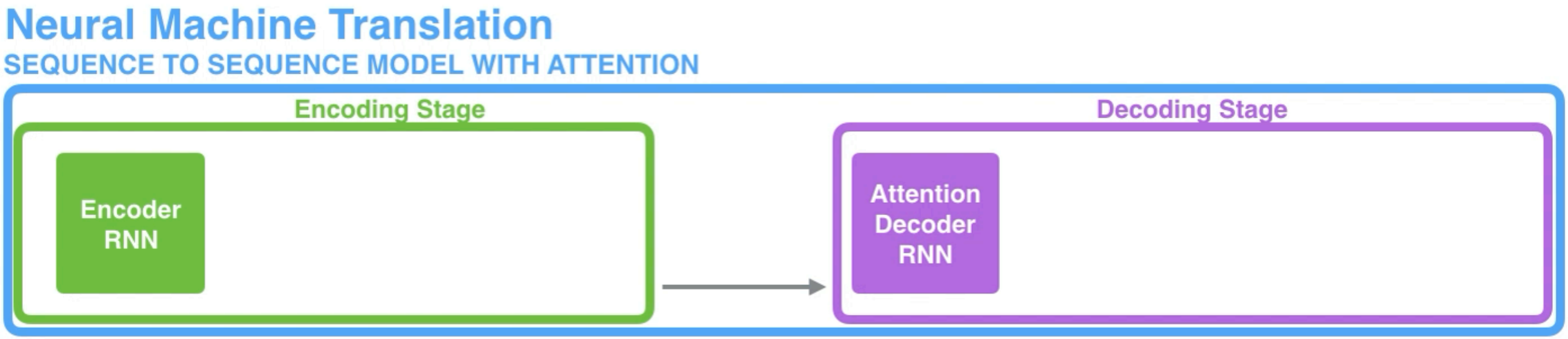
Sequence to Sequence mapping



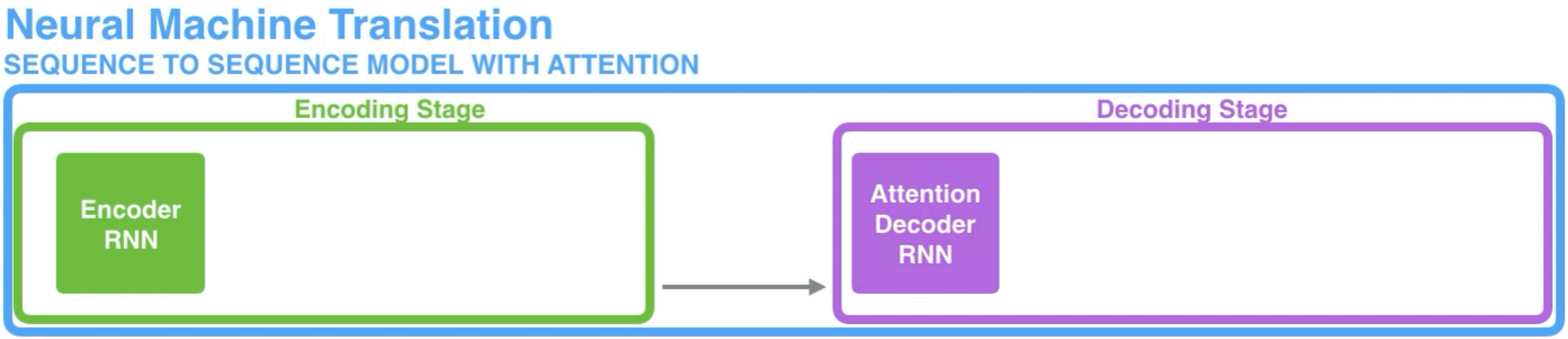
Sequence to Sequence mapping



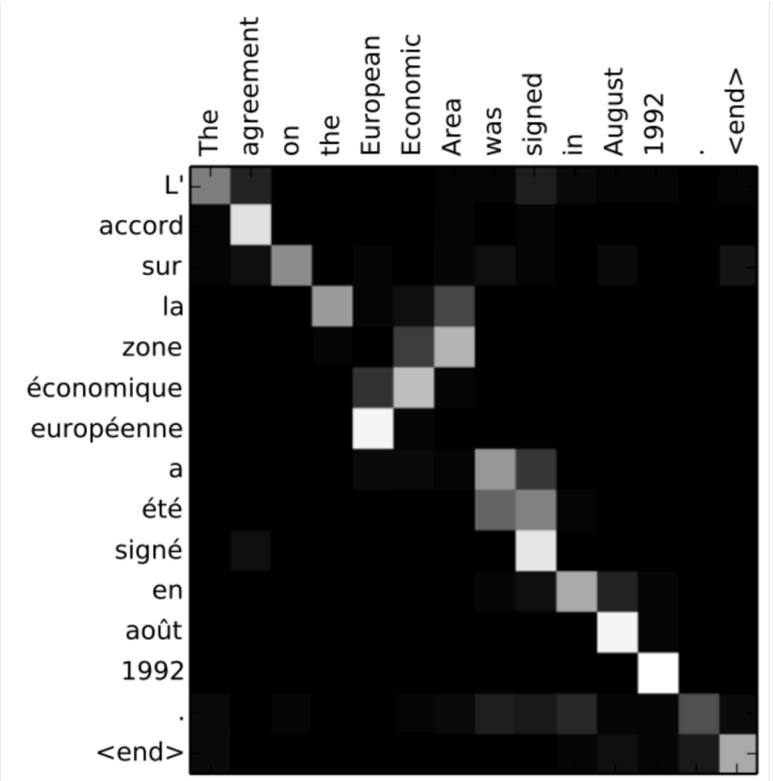
Sequence to Sequence mapping



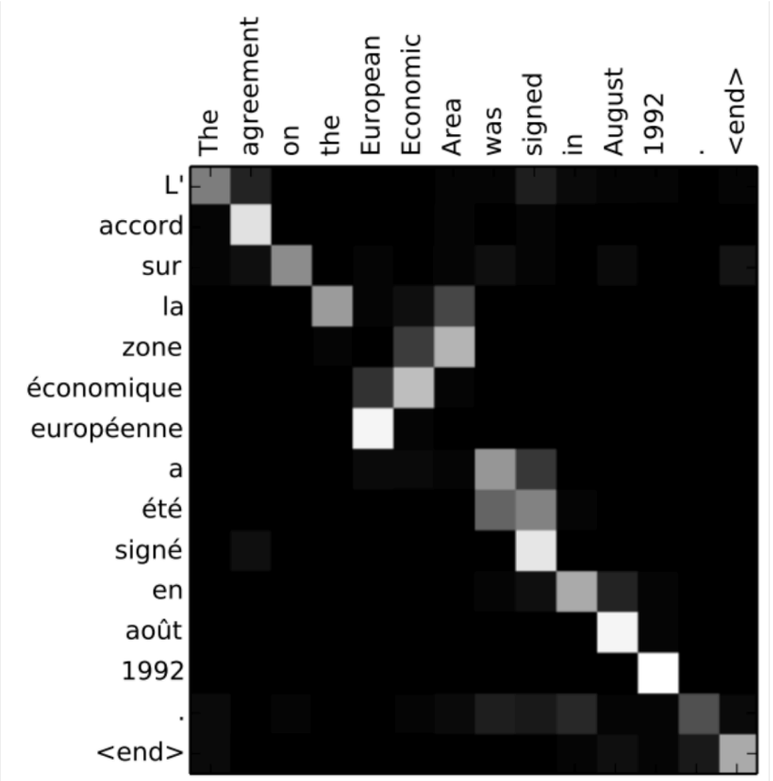
Sequence to Sequence mapping



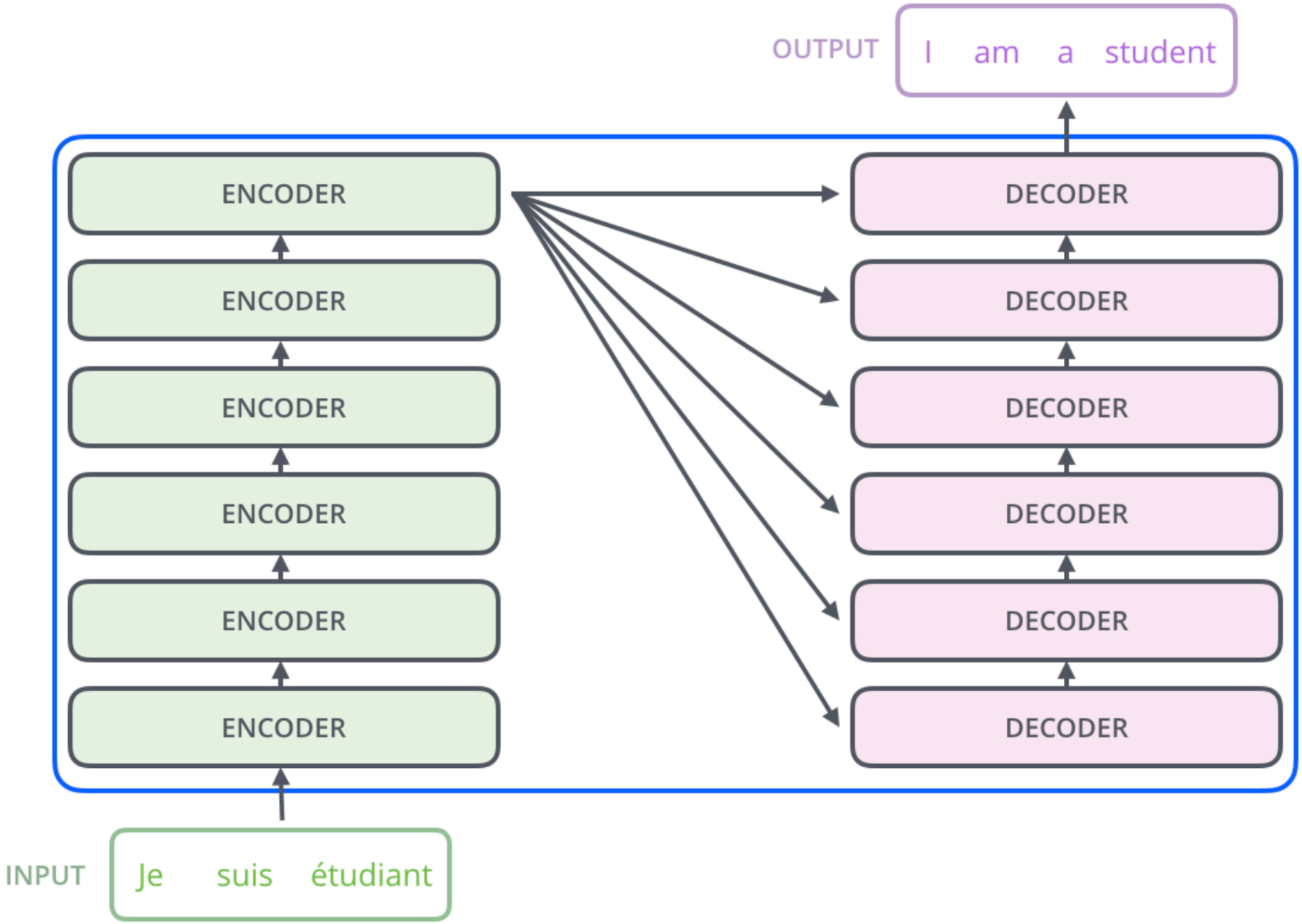
Sequence to Sequence mapping



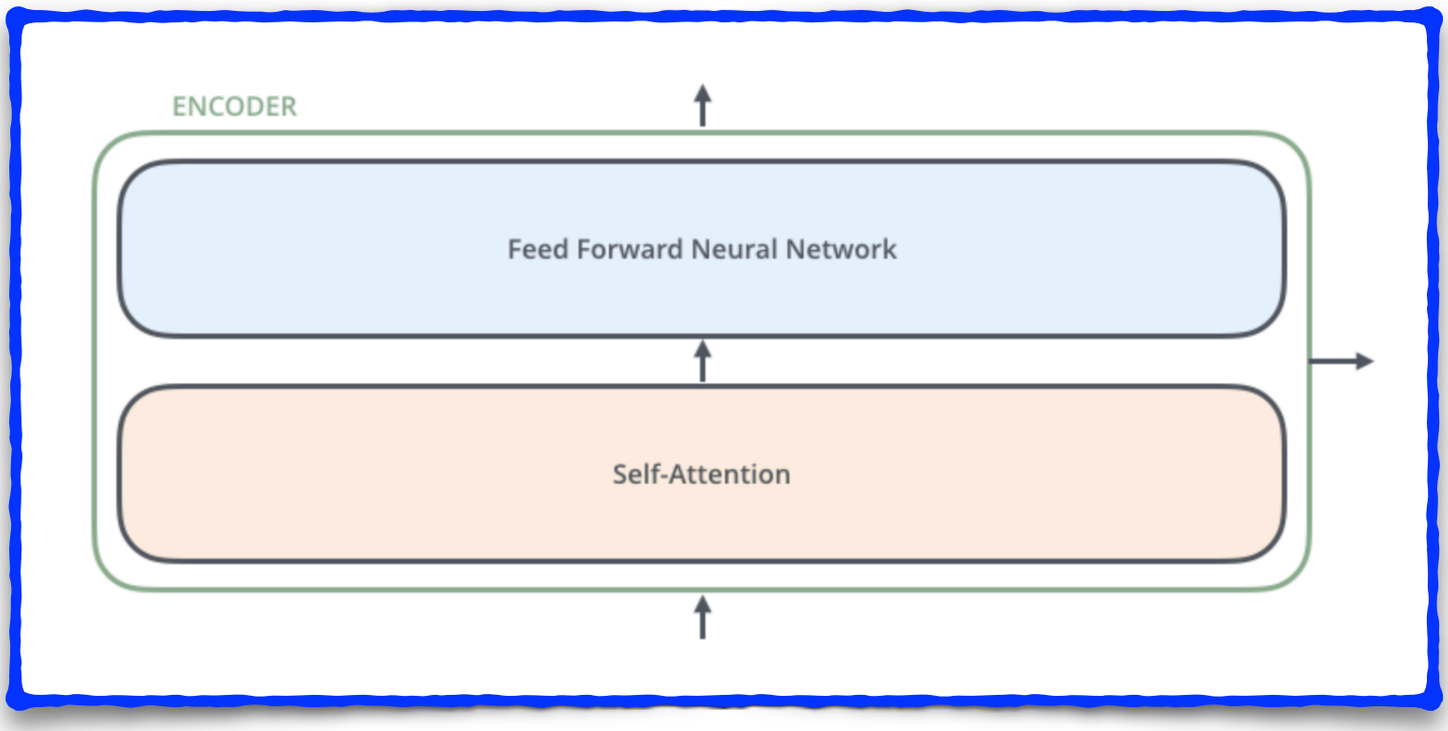
Sequence to Sequence mapping



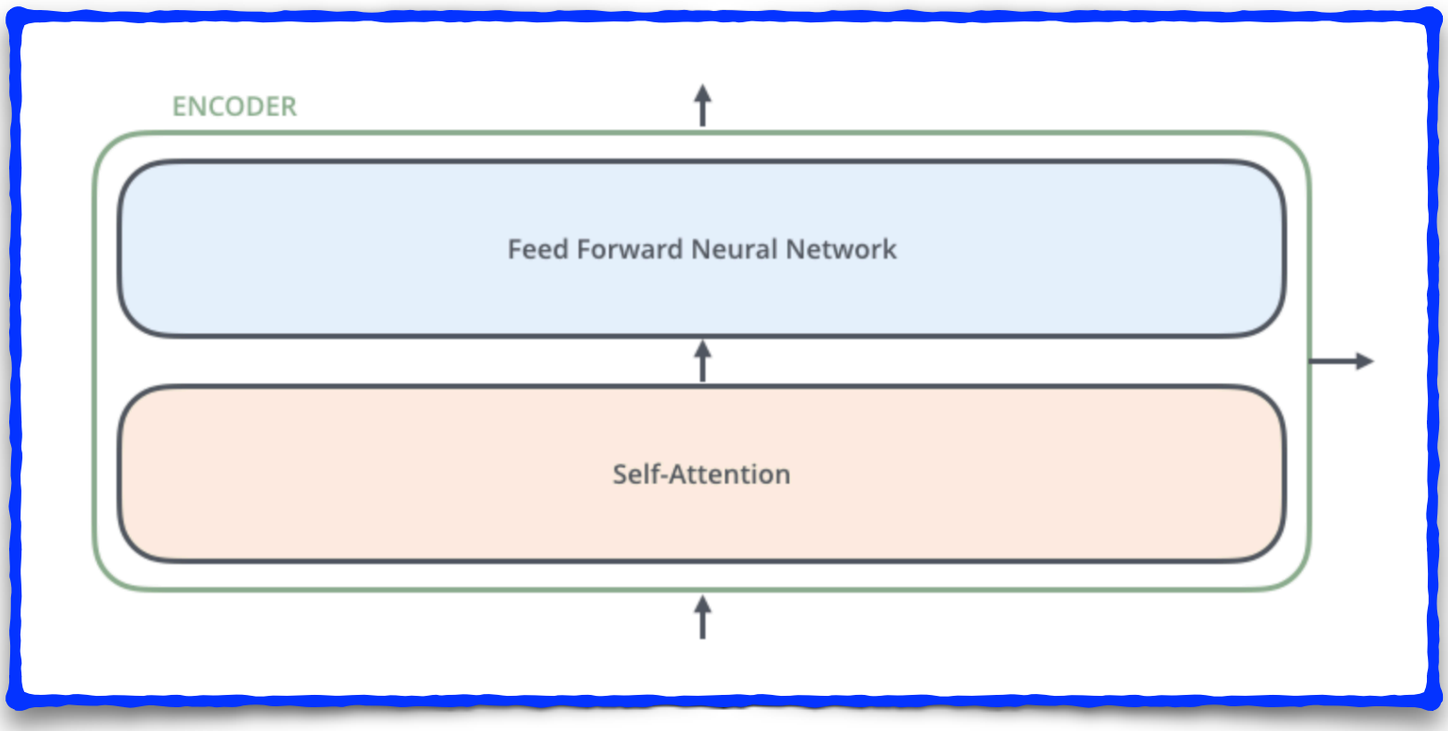
Sequence to Sequence mapping

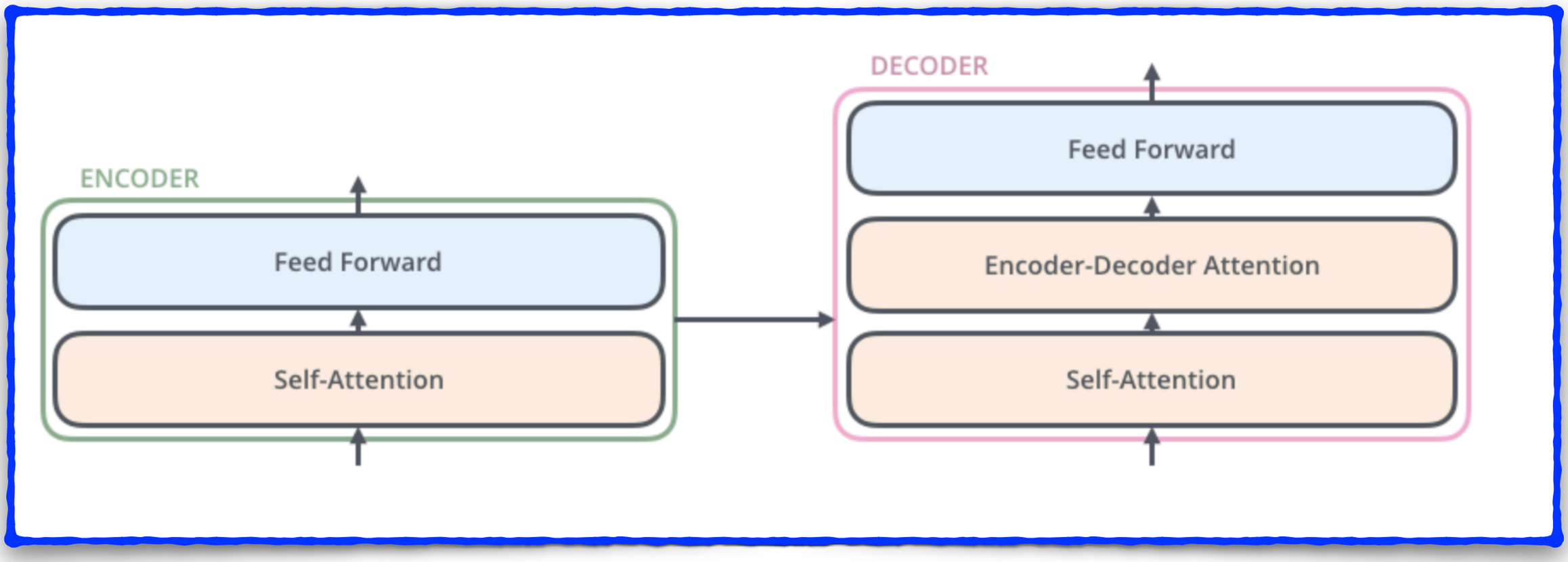


Encoder feeds the decoder

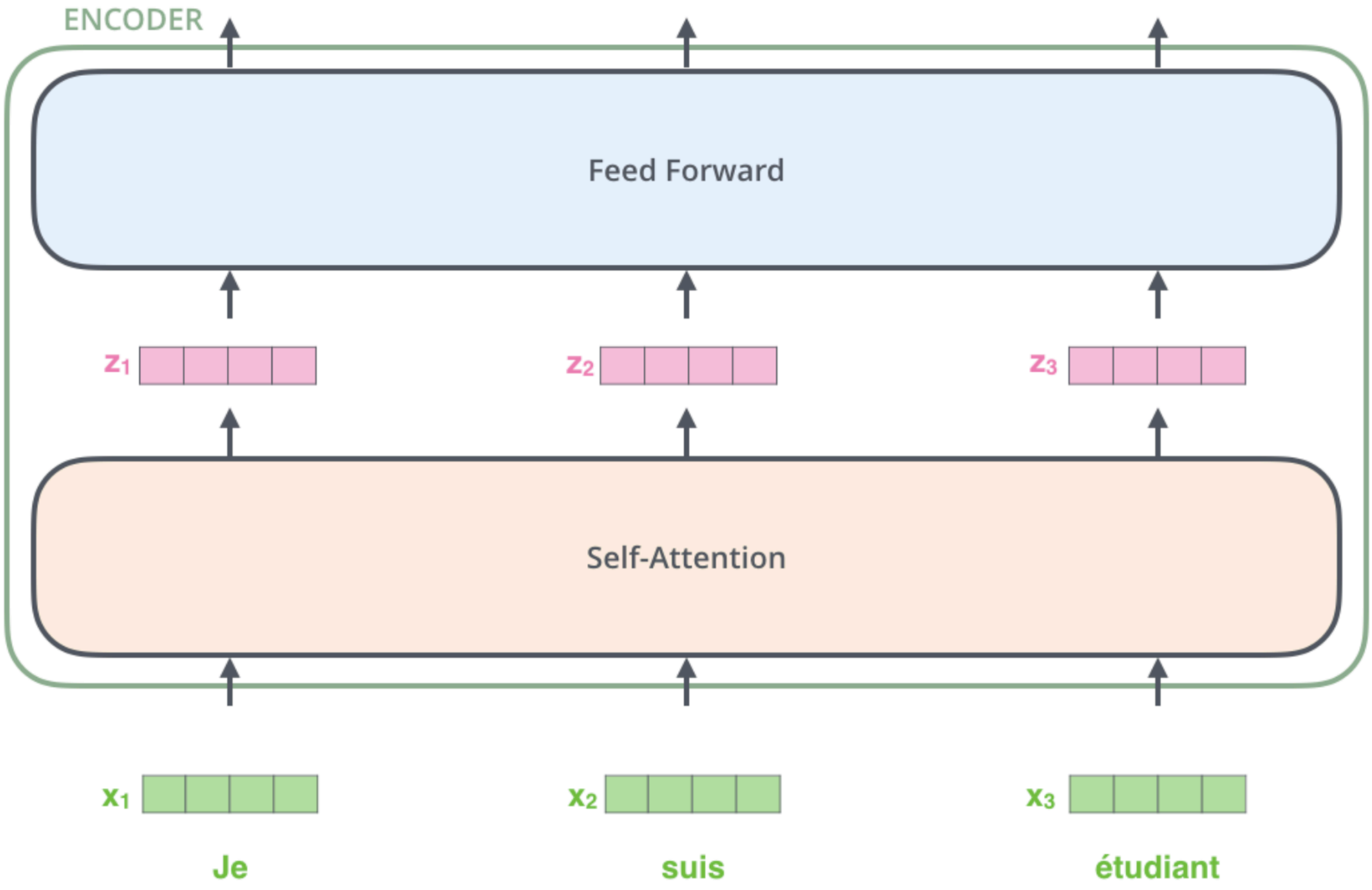


Encoder feeds the decoder



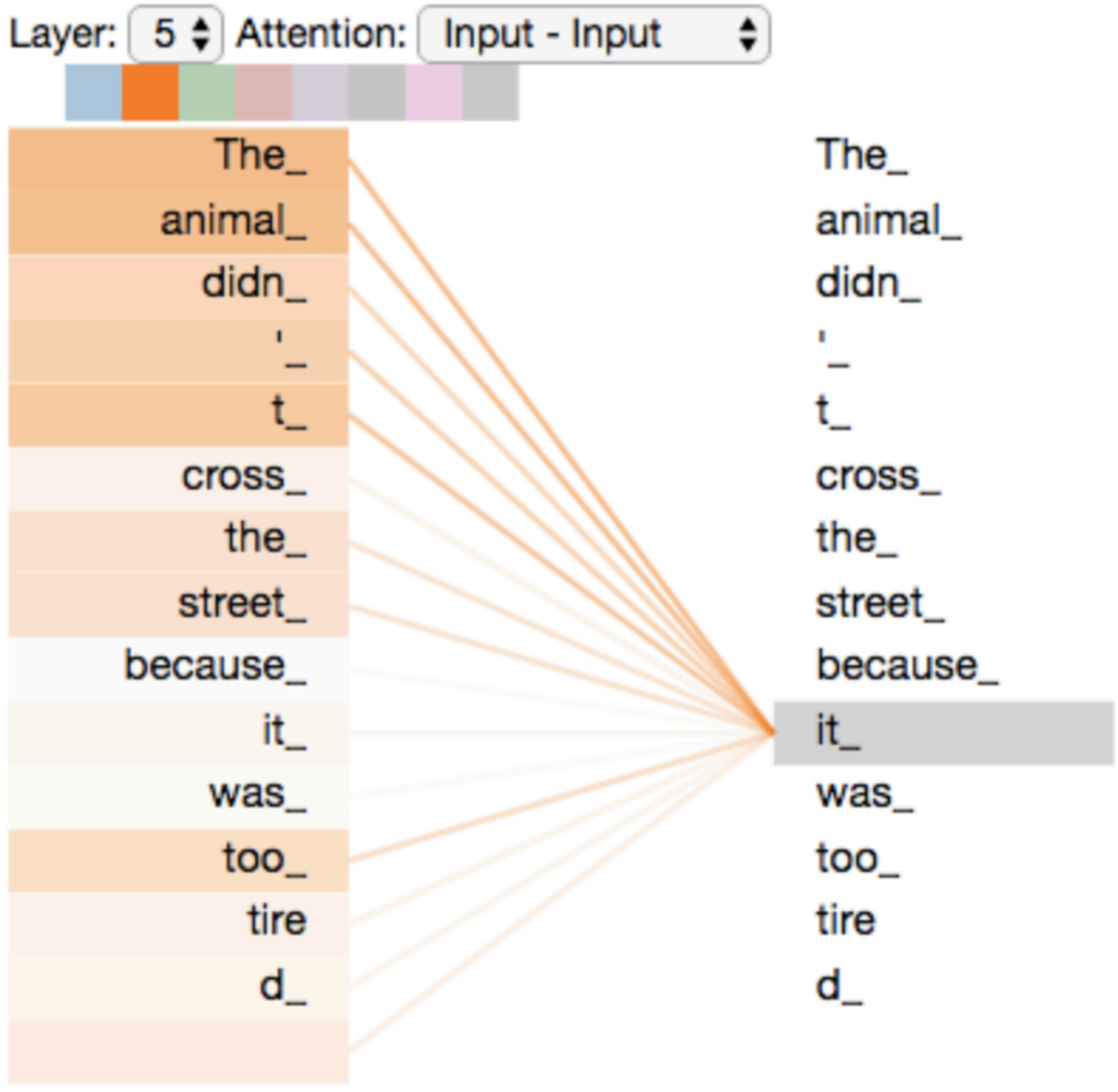


Encoder feeds the decoder

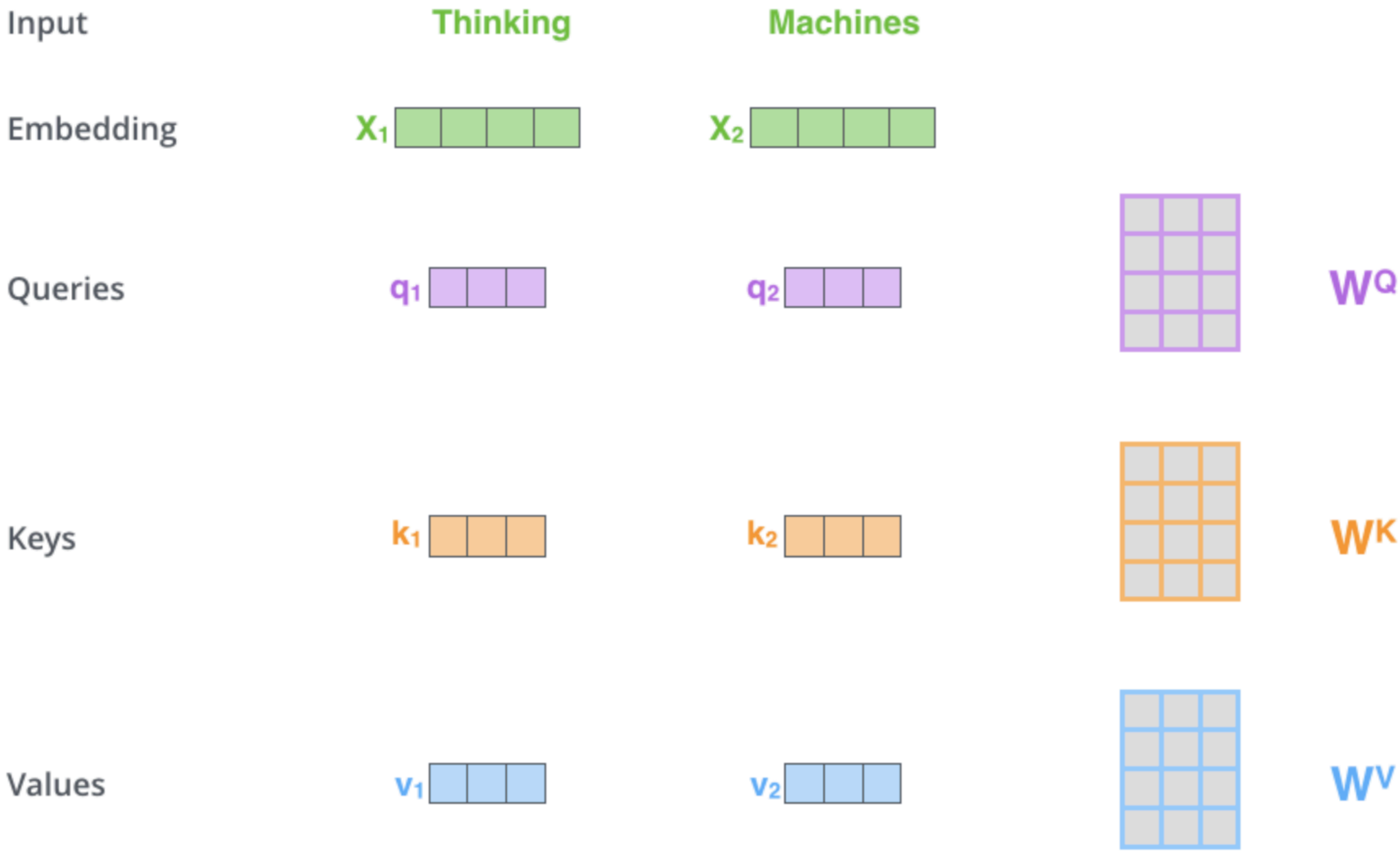


The concept of self-attention

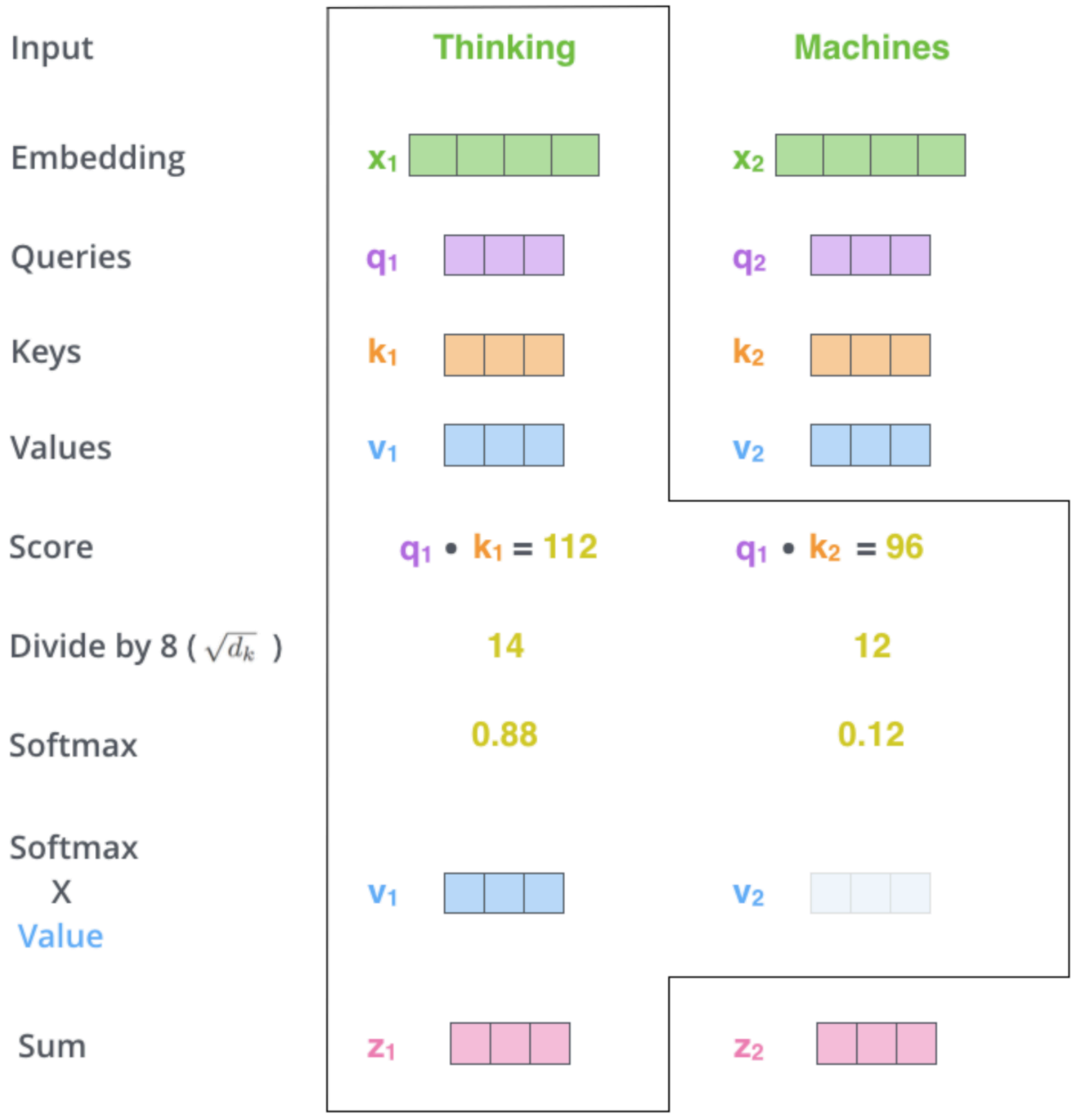
"The animal didn't cross the street because it was too tired"



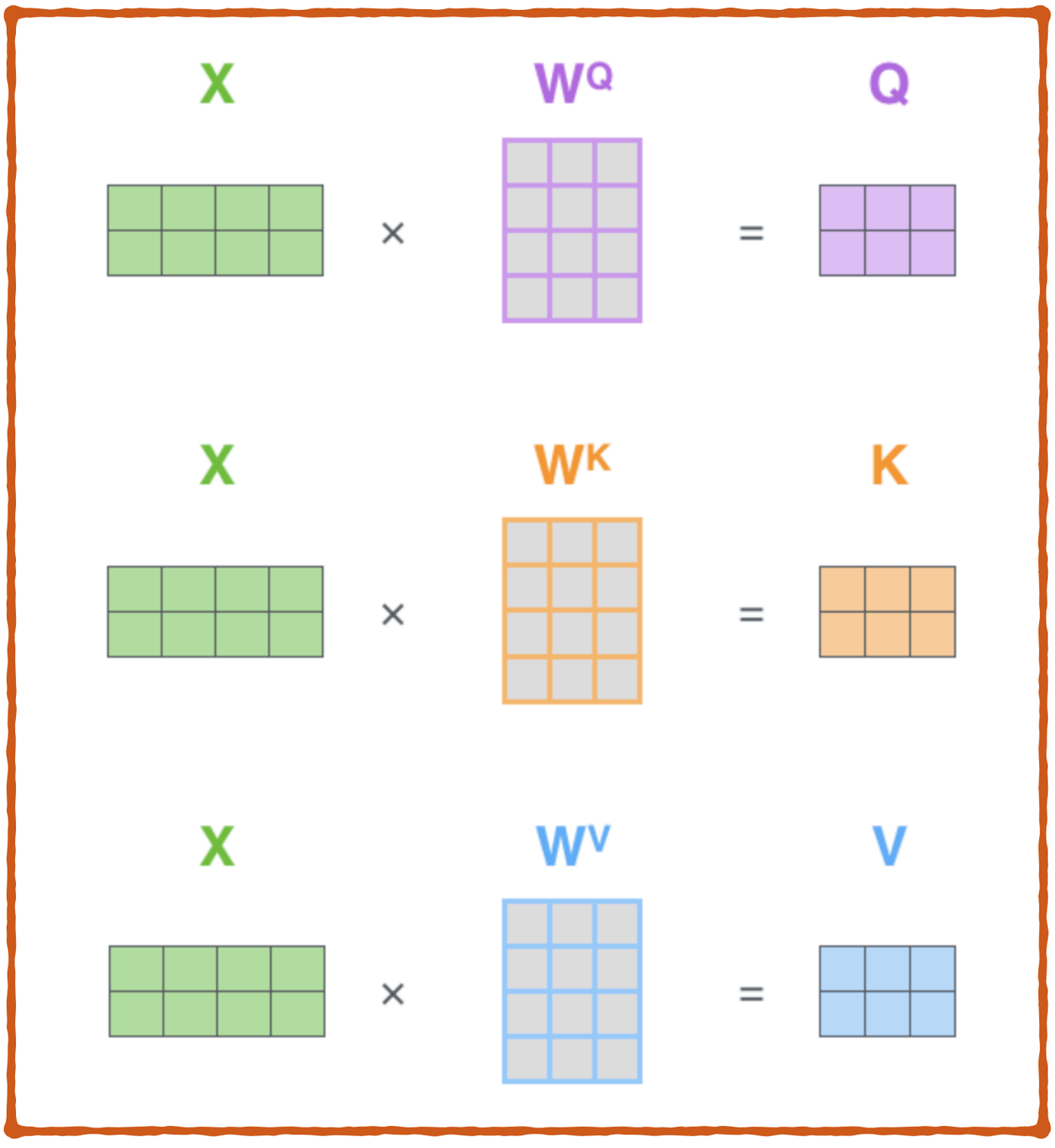
How self attention works?

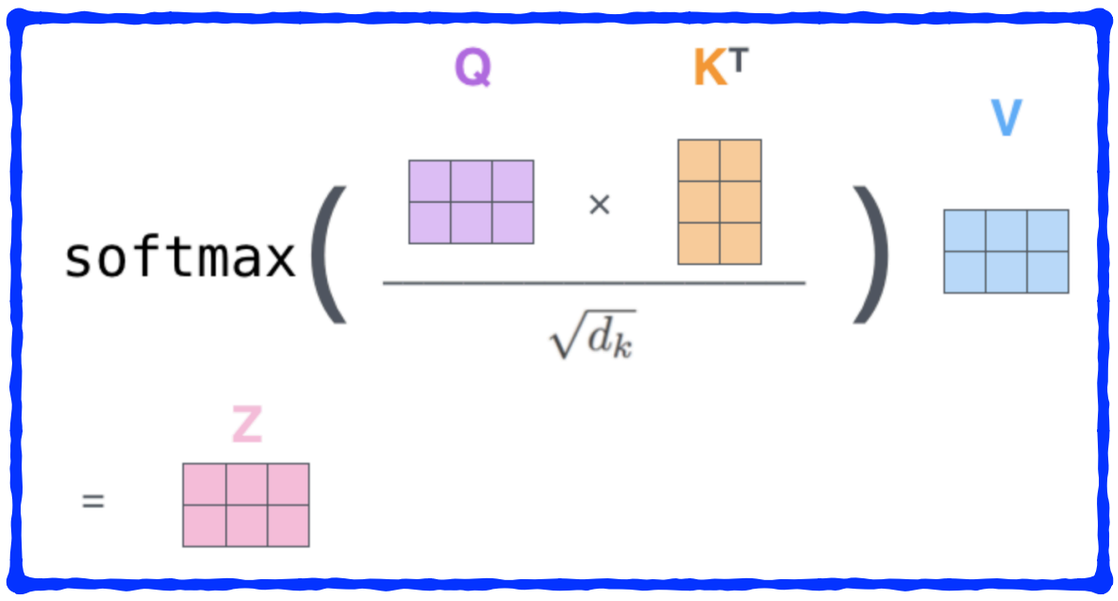


How self attention works?

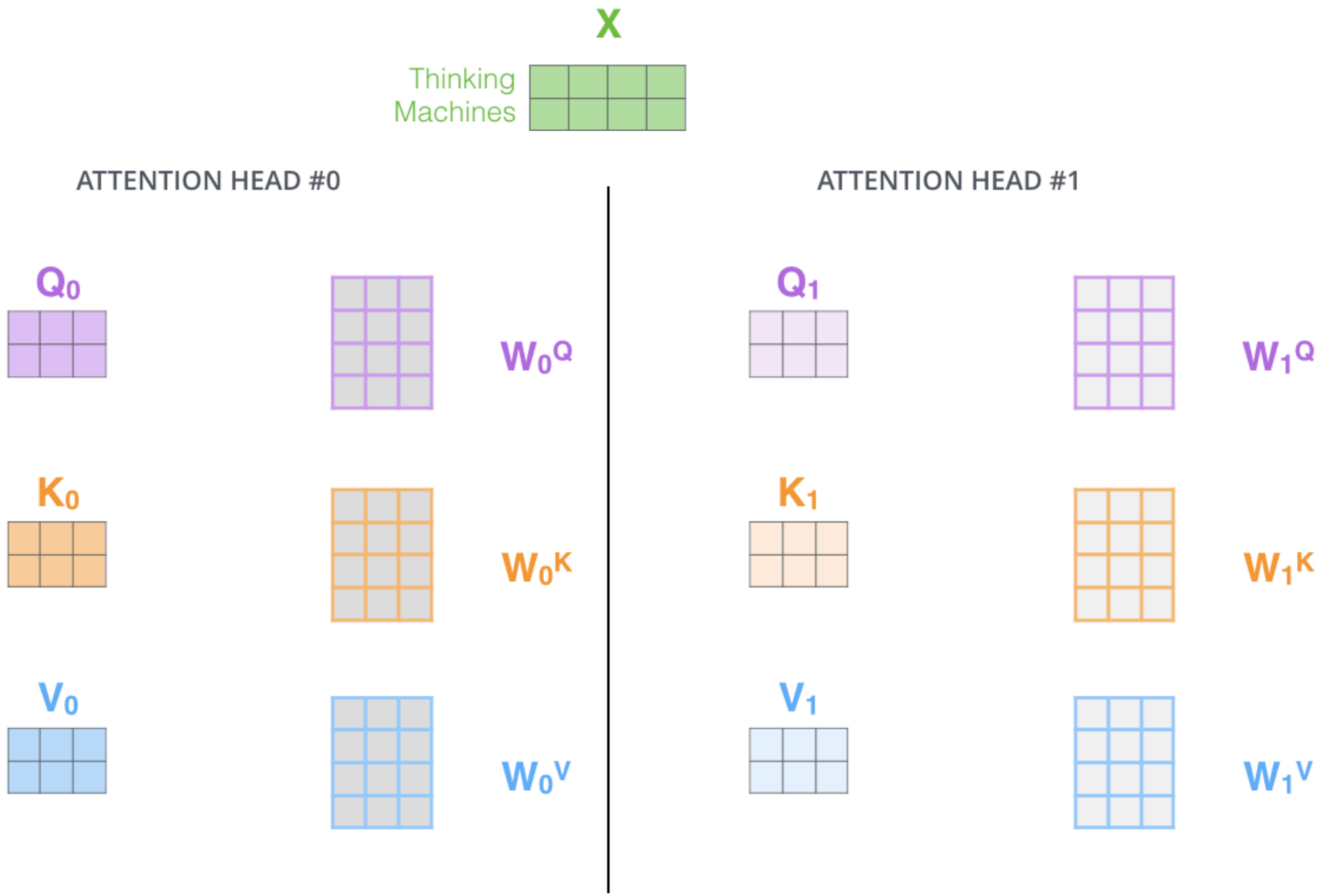


Computing the attention values





Computing the multi-head attention



Computing the multi-head attention

1) This is our input sentence*

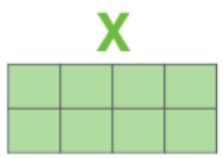
2) We embed each word*

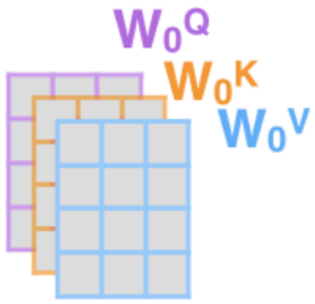
3) Split into 8 heads. We multiply X or R with weight matrices

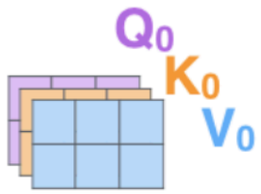
4) Calculate attention using the resulting Q/K/V matrices

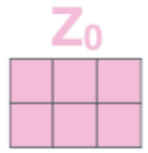
5) Concatenate the resulting Z matrices, then multiply with weight matrix W^O to produce the output of the layer

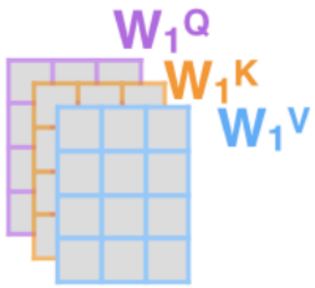
Thinking Machines

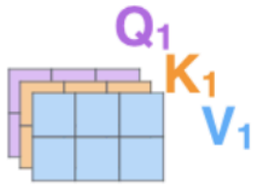


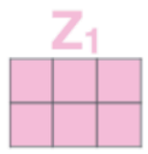


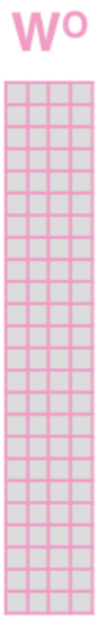




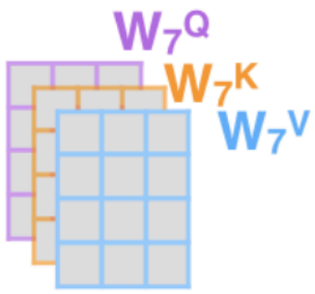


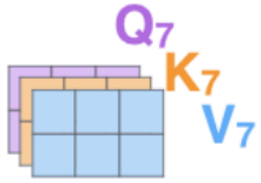


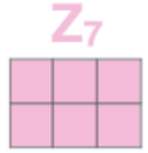


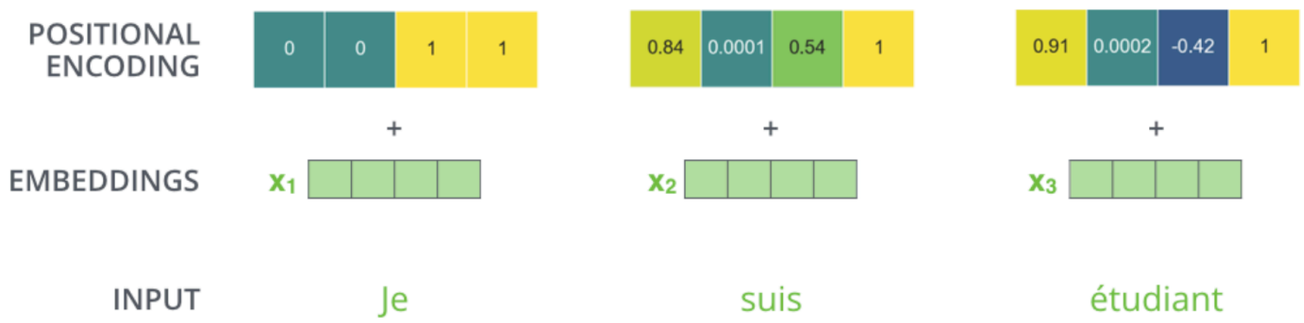


* In all encoders other than #0, we don't need embedding. We start directly with the output of the encoder right below this one



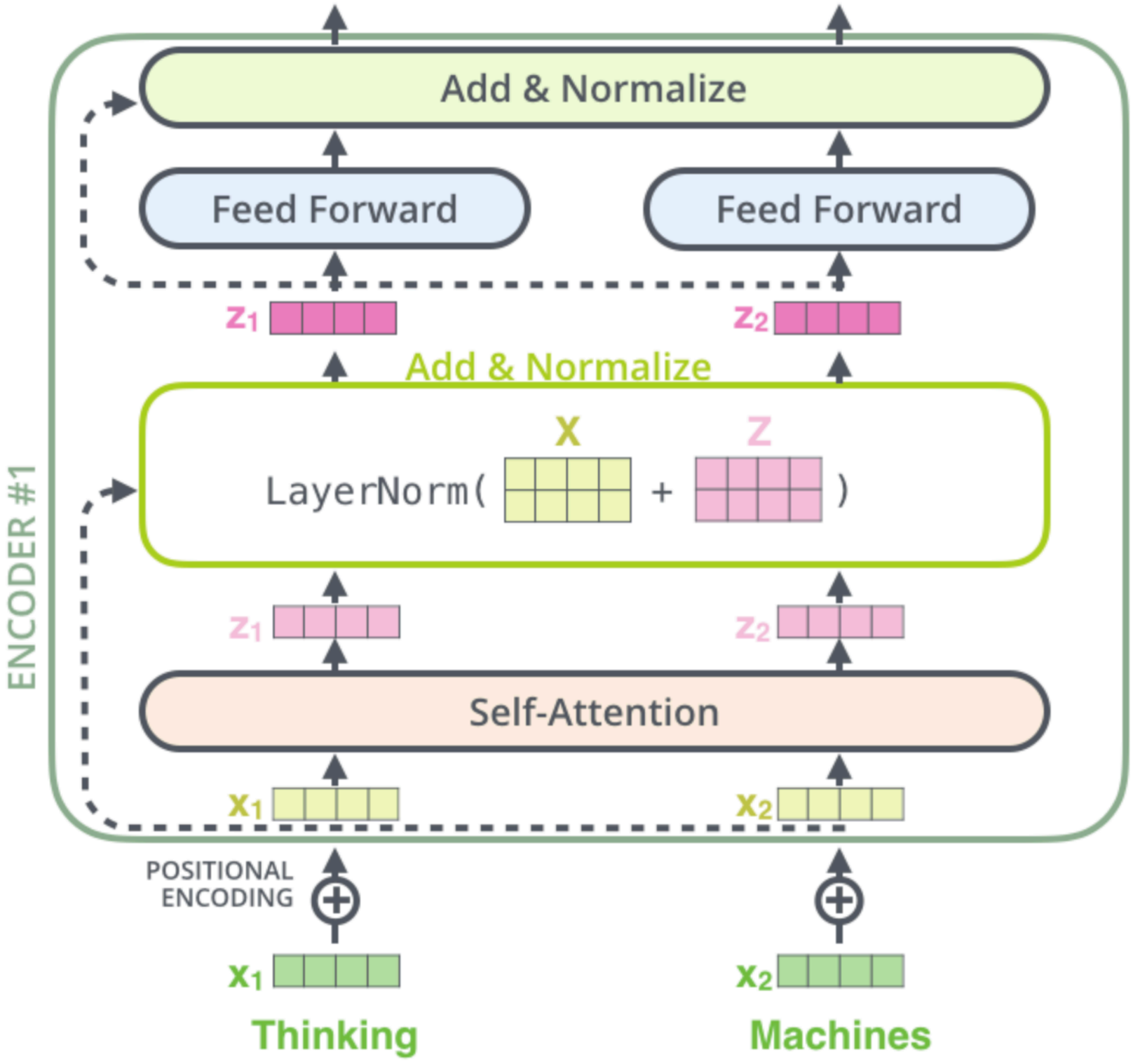




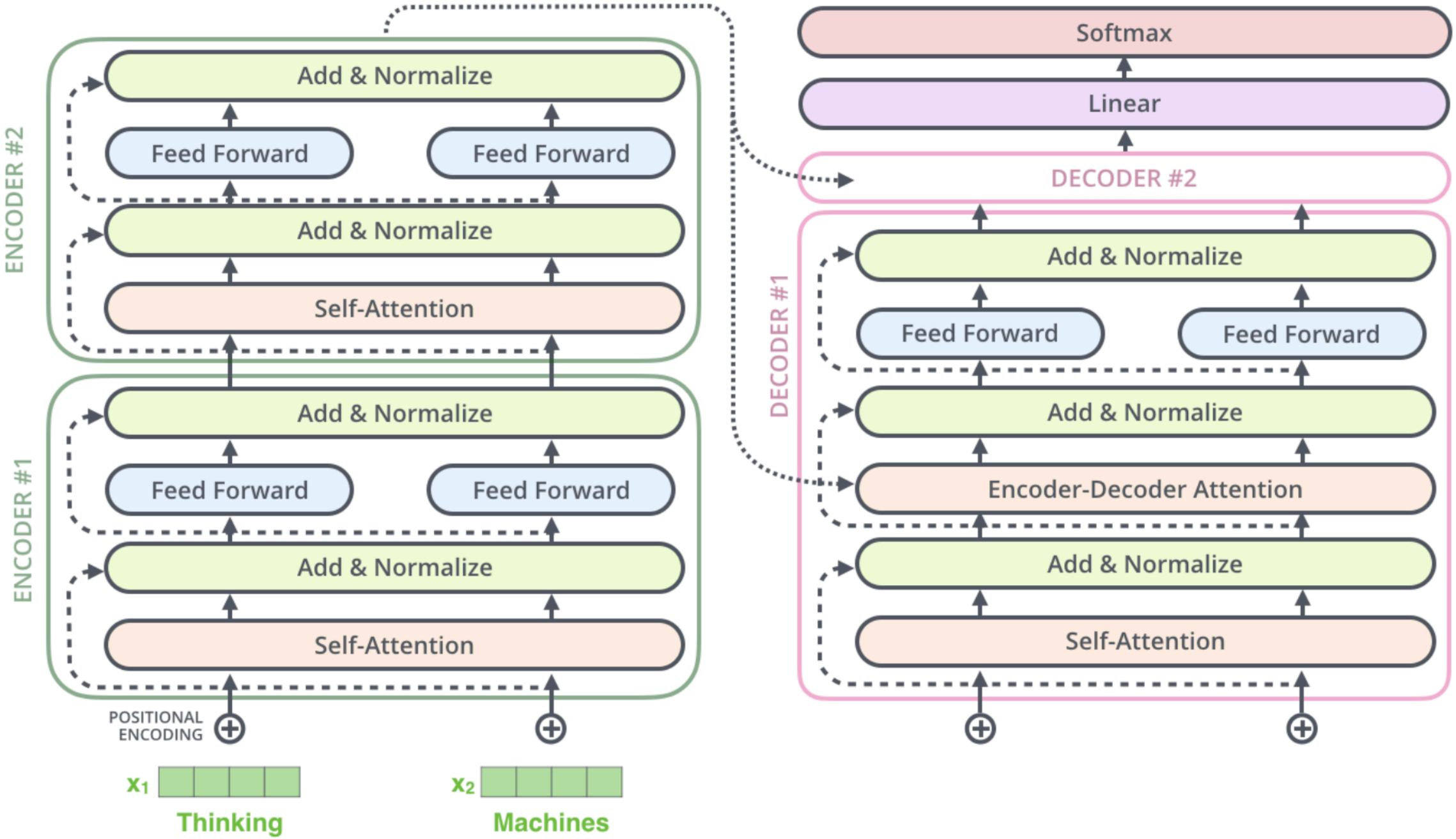


Add the time stamp to ensure positional encoding

Let's see how the encoder looks



The simplest encoder-decoder net



Transformer architecture