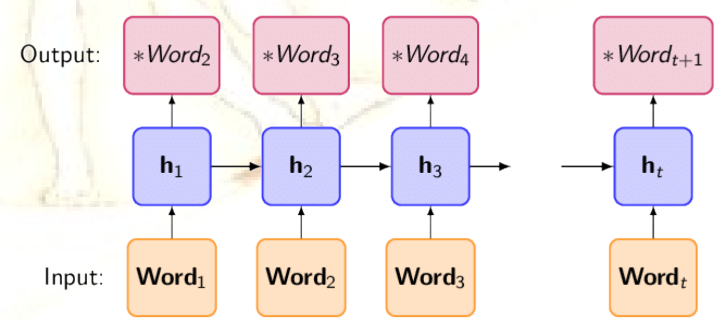
Neural Translation - Transformer

Pattern Recognition & Machine Learning Laboratory
Ji-Sang Hwang, Aug 5, 2021

Attention is All You Need [A. Vaswani et al., 2017] (1/9)

- **Introduction**
  - Recurrent model
    - Having been firmly established as state-of-the-art approach in sequence modeling and transduction problems
    - Factoring computation along the symbol positions of the input and output sequences
    - Precluding parallelization within training examples because of inherently sequential nature
  - Attention mechanisms
    - Becoming an integral part of compelling sequence modeling and transduction models in various tasks
    - Allowing modeling of dependencies without regard to their distance in the input or output sequences [*Y. Kim*, 2017]
- **Goal**
  - A model architecture eschewing recurrence
  - Relying entirely on an attention mechanism to draw global dependencies between input and output



Architectures of RNN

Y. Kim et al., "Structured attention networks", *ICLR*, 2017

Attention is All You Need [A. Vaswani et al., 2017] (2/9)

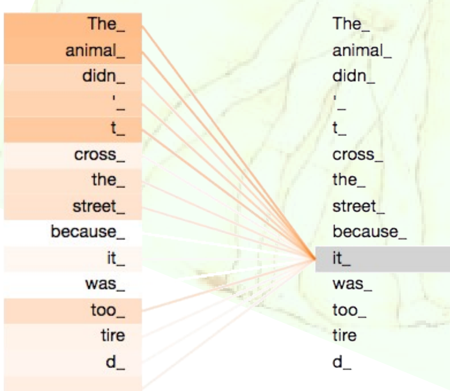
- **Background**
  - **Reducing sequential computation**
    - **Using convolutional neural networks (CNN) as basic building block**
    - **Computing hidden representations in parallel for all input and output positions**
    - **The number of operations required to relate signals from two arbitrary input or output positions grows in the distance between postions**
      - **ConvS2S : Linearly [*J. Gehring et al*., 2017]**
      - **ByteNet : Logarithmically [*N. Kalchbrenner et al*., 2017]**
  - **Self-Attention [*Z. Lin et al*., 2017]**
    - **Relating different positions of a single sequence in order to compute a represen-tation of the sequence**
    - **Used successfully in a variety of tasks**
  - **End-to-end memory networks [*S. Sukhbaatar et al*., 2015]**
    - **Based on a recurrent attention mechanism instead of sequence-aligned recur-rence**
    - **Shown to perform well on simple-language question answering and language modeling tasks**

J. Gehring et al., "Convolutional sequence to sequence learning", *ICLR*, 2017.
N. Kalchbrenner et al., "Neural machine translation in linear time", *arXiv*, 2016.
Z. Lin et al., "A structured self-attentive sentence embedding", *arXiv*, 2017.
S. Sukhbaatar et al., "End-to-end memory networks", *arXiv*, 2015.

Attention is All You Need [A. Vaswani et al., 2017] (3/9)

- **Method**
  - **Self-Attention**
    - **Allowing each word to look at other positions in the input sequence for clues that can help lead to a better encoding for this word**
    - **Baking the 'understanding' of other relevant words into the one currently pro-cessing**
    - **Scaled dot-product Attention**

- : A vector of Queries (particular output)
- : A vector of keys (input sequence)
- : A vector of values (multiplying weights with input sequence)
- : A demention of queries and keys



Example of understanding

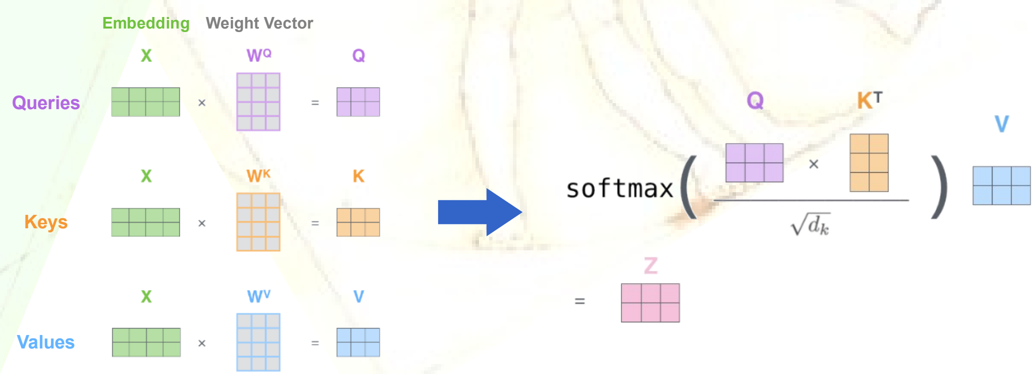
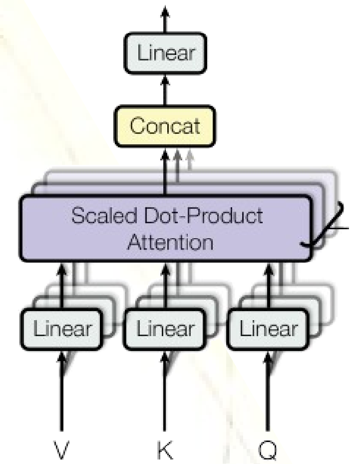


Image of Calculation Scaled-Dot-Product Attention

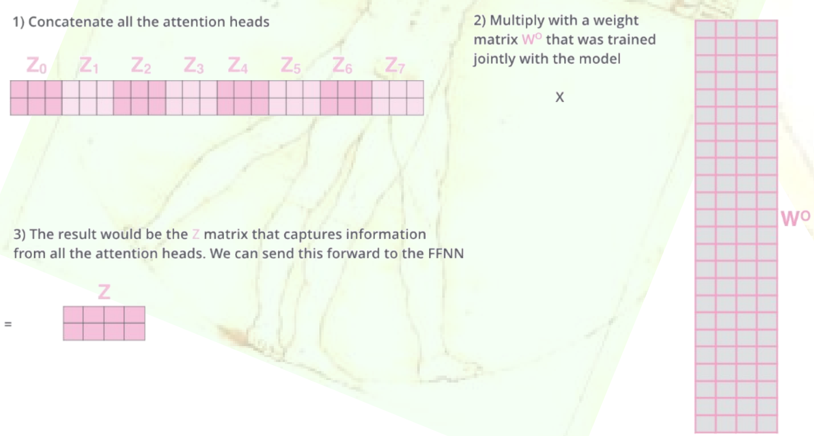
Attention is All You Need [A. Vaswani et al., 2017] (4/9)

- **Multi-Head Attention**
  - Expanding the model's ability to focus on different positions.
  - Giving the attention layer multiple 'representation subspaces'
    - Having multiple sets of query, key and value weight matrices (8 matrices in Transformer)

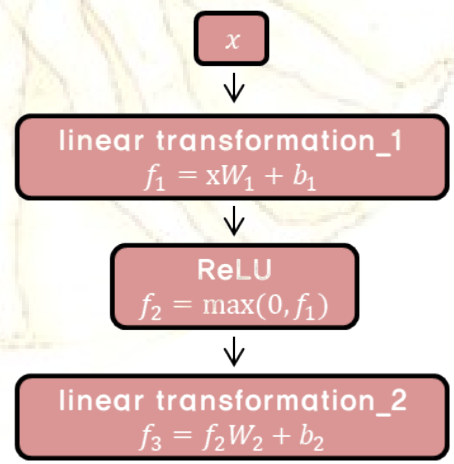


Architecture of Multi-Head Attention

- **Position-wise Feed-Forward Networks (FFN)**
  - Appling to each position separately and identically



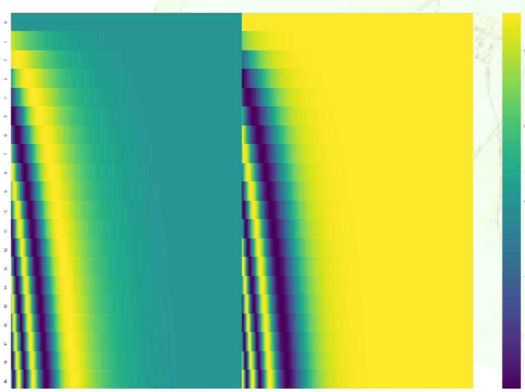
Sequence of Multi-Head Attention



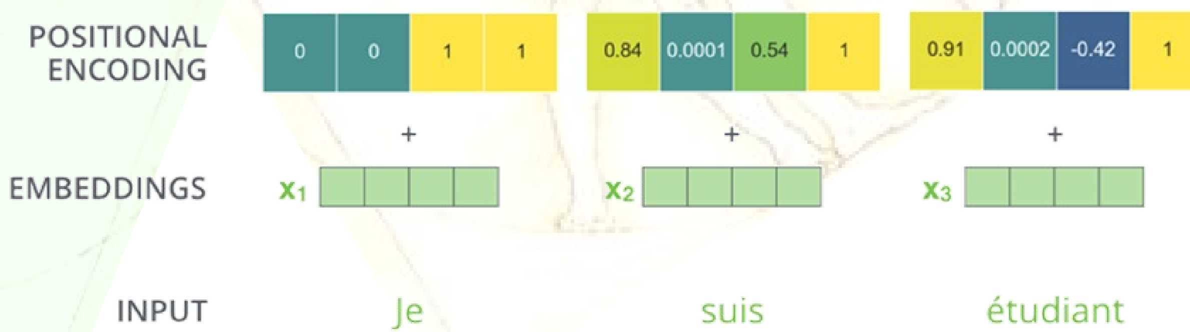
Architecture of Position-wise FFN

Attention is All You Need [A. Vaswani et al., 2017] (5/9)

- **Embeddings**
  - Using learned embeddings to convert the input tokens and output tokens to vectors of dimension
  - Using usual learned linear transformation and softamx function
    - Converting the decoder output to predicted next-token probabilits
- **Positional encoding**
  - Having the same dimension as the embeddings
  - Using sine and cosine functions of different frequencies
    - , : position, : the dimension



Example Positional encoding

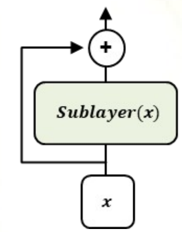


Example of Embeddings and Positional encoding

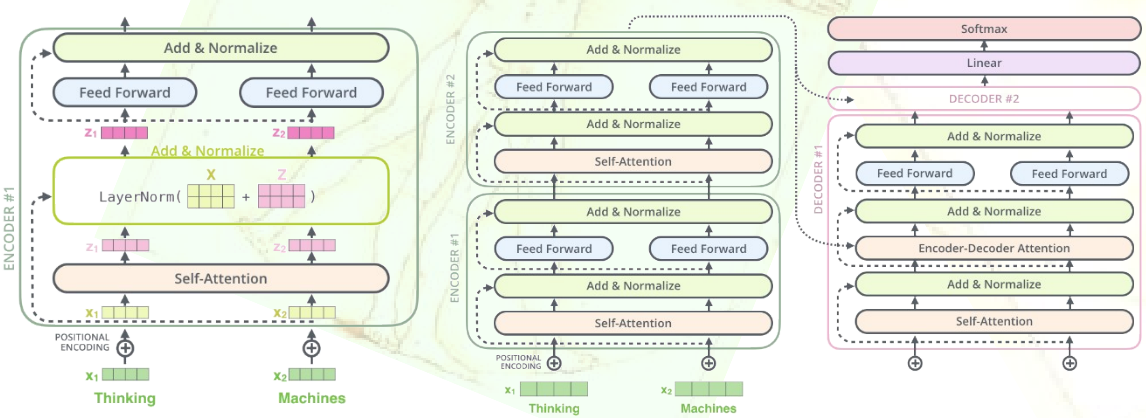
Attention is All You Need [A. Vaswani et al., 2017] (6/9)

Encoder

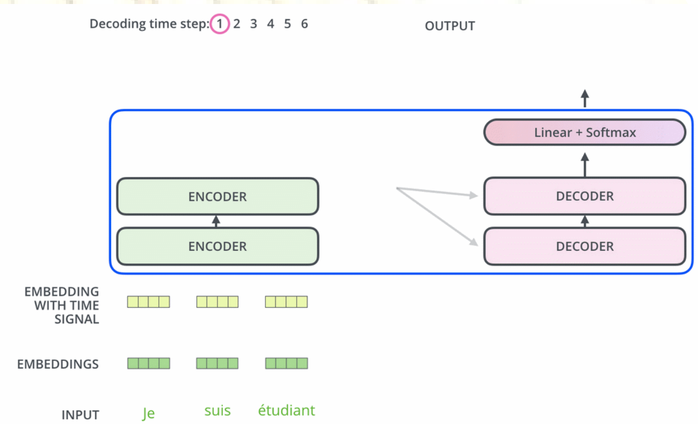
- Stack of identical layers
- Each layer has 2 sub-layers (Multi-head self-attention mechanism & fully connected feed-forward network)
- Employing a residual connection around each of two sub-layers, following by layer normalization
  - Residual Connection
  - Layer Nomarlization



Example of Residual Connection



Detail architecture of Encoder to Decoder

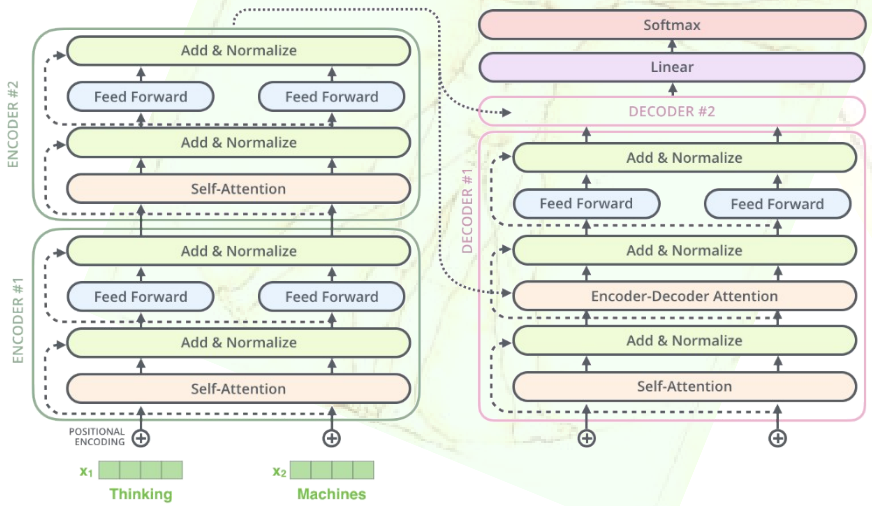


Example of Encoder to Decoder

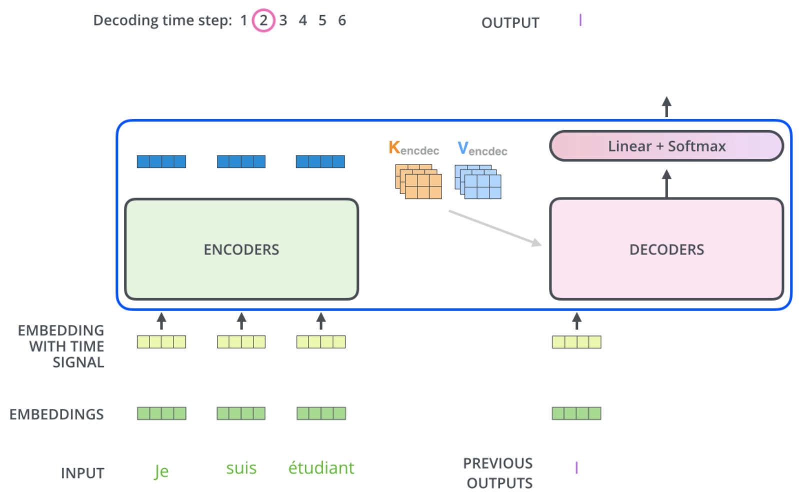
Attention is All You Need [A. Vaswani et al., 2017] (7/9)

Decoder

- Composing of a stack of identical layers
- Masked Multi-Head Attention
  - Modifing the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions
  - Ensuring predictions for position can depend only on the known outputs at positions less than
- Encoder-Decoder Attention
  - Allowing every position in the decoder to attend over all positions in the input sequence



Detail architecture of Encoder

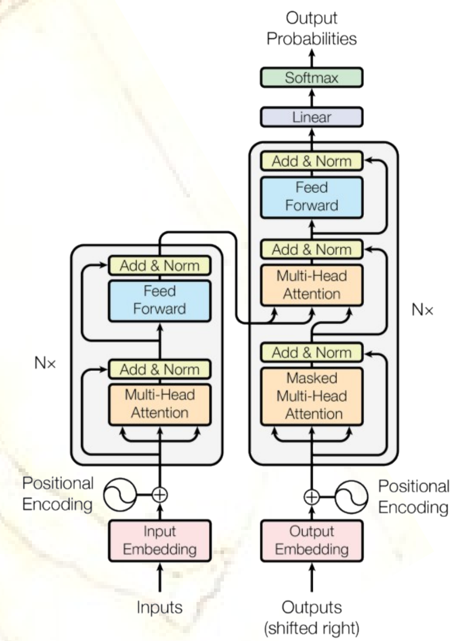


Example of Decoder

Attention is All You Need [A. Vaswani et al., 2017] (8/9)

➢ Training Transformer

- **Training Data**
  - Training Data
    - » Workshop on Statistical Machine Translation (WMT) 2014 English-German dataset consisting of about 4.5 million sentence pairs
    - » WMT 2014 English-French dataset consisting of about 36 million sentence pairs
- **Optimizer**
  - Using Adam
- **Learning Rate**



Architecture of Transformer

- **Regularized by Residual Dropout and Label Smoothing**
  - Dropout
  - Label Smoothing
    - » Hurting perplexity, as the model learns to be more unsure, but improving accuracy and BiLingual Evaluation Understudy (BLEU) score.

Attention is All You Need [A. Vaswani et al., 2017] (9/9)

- **Result**
  - **Reason of using Self-Attention**
    - **Reducing total computational complexity per layer**
    - **Increasing amount of computation that can be parallelized**
    - **Learning path length between long-range dependencies in the network**

Table of Complexity

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)	$O(r \cdot n \cdot d)$	$O(1)$	$O(n/r)$

» : Sequence length, : Representation dimension

» : Kernel size, : Size of neighborhood

- **Conclusion**
  - **Trained significantly faster than architectures based on recurrent or convolutional layers**

Table of Machine Performance in WMT 2014

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**	$2.3 \cdot 10^{19}$	

Visualization of Machine Translation