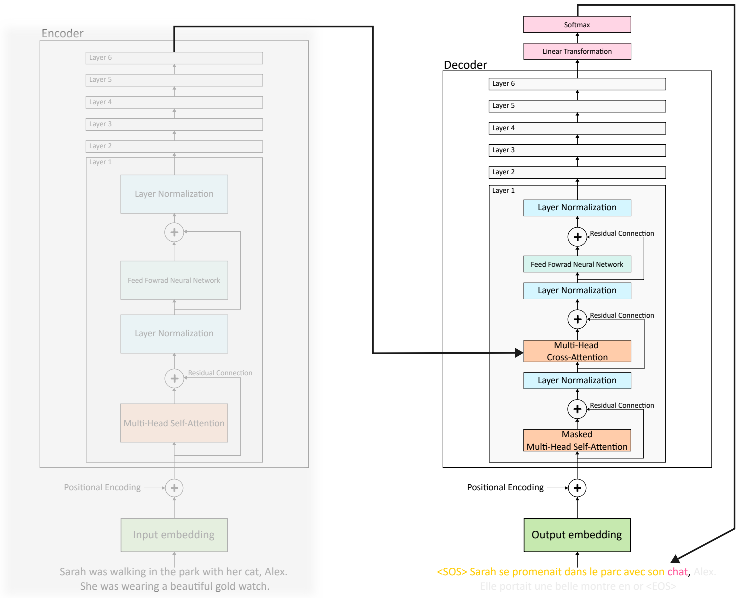
Transformers (Part 6)

Dr. Alireza Aghamohammadi

Transformer Decoder Architecture



Masked Multi-Head Self Attention

- When the decoder is processing the token at position t, it should not have access to tokens at positions $t+1$ and beyond.
- To achieve this, we can use a mask matrix M where entries that should not be attended to are set to $-\infty$, and other entries are set to 0:

$$A = \text{softmax}\left(M + \frac{K^T \cdot Q}{\sqrt{d}}\right)$$

- For example, the following mask matrix is commonly used in transformers:

$$M = \begin{bmatrix} 0 & -\infty & -\infty & \dots & -\infty \\ 0 & 0 & -\infty & \dots & -\infty \\ 0 & 0 & 0 & \dots & -\infty \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}$$

Cross Attention

- To make our inputs learnable, we define:

$$q = W_q \gamma_i \quad k = W_k \zeta_i \quad v = W_v \zeta_i$$

- This gives us m queries $Q = [q_1, q_2, \cdots, q_m]$, τ keys $K = [k_1, k_2, \cdots, k_\tau]$, and τ values $V = [v_1, v_2, \cdots, v_\tau]$:

$$Q \in \mathbb{R}^{d \times m}$$
$$K, V \in \mathbb{R}^{d \times \tau}$$

- The attention matrix is calculated as:

$$A = \mathsf{softmax}\left(\frac{K^T \cdot Q}{\sqrt{d}}\right) \in \mathbb{R}^{\tau \times m}$$

- The normalization factor $\sqrt{d}$ helps to prevent the dot products from becoming too large, which can lead to very small gradients.
- The softmax function ensures that the attention weights are positive and sum to 1.
- Finally, the hidden representation is defined as:

$$H = VA \in \mathbb{R}^{d \times m}$$