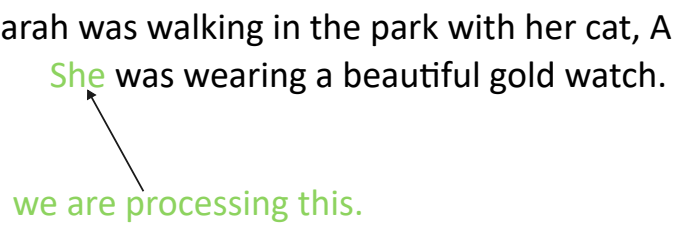
Transformers (Part 4)

Dr. Alireza Aghamohammadi

Self Attention Mechanism

- Given an input sequence $X = (x_1, x_2, \cdots, x_m)$.
- When processing the token x_t (e.g., "She"), we determine how much attention to give to each other token x_i (e.g., "Sarah", "cat", "Alex").



- Each token x_i is a vector: $x_i \in \mathbb{R}^n$, so $X \in \mathbb{R}^{n \times m}$.
- The attention weight of token x_t to token x_i is formalized as:

$$h_t = \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_m x_m \quad \text{where} \quad \alpha_i \geq 0$$

- In matrix form, this is written as $h_t = X a_t \in \mathbb{R}^n$, where $a_t = [\alpha_1, \alpha_2, \cdots, \alpha_m]^T$.

Deterministic Approach for Attention Weights

- Our goal is to find the attention weights vector $a_t = [\alpha_1, \alpha_2, \cdots, \alpha_m]^T$. How can we achieve this?
- Let's simplify the problem and solve it deterministically without learning.
- The similarity between two vectors x_t and x_i can be computed as the dot product $x_t \cdot x_i$. We can use this idea.
- For each token x_t, we compute the attention weights vector $a_t \in \mathbb{R}^m$:

$$a_t = \text{softmax}\left(X^T \cdot x_t\right) \in \mathbb{R}^m$$

- The softmax function ensures that the attention weights are positive and sum to 1.
- We define the matrix $A \in \mathbb{R}^{m \times m}$, where each column corresponds to the attention weights vector:

$$A = [a_1, a_2, \cdots, a_m]$$

- Finally, the hidden representation is defined as:

$$H = XA \in \mathbb{R}^{n \times m}$$

- Note that this similarity-based attention does not consider the context of the sentence. Therefore, we need to incorporate parameters to learn the context as well.

Learnable Queries, Keys, and Values

- To make our inputs learnable, we define:

$$q = W_q x_i \quad k = W_k x_i \quad v = W_v x_i$$

- Here, $W_q, W_k, W_v \in \mathbb{R}^{d \times n}$, so $q, k, v \in \mathbb{R}^d$.
- This gives us m queries $Q = [q_1, q_2, \cdots, q_m]$, m keys $K = [k_1, k_2, \cdots, k_m]$, and m values $V = [v_1, v_2, \cdots, v_m]$:

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

- The attention matrix is calculated as:

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

- 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:

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