23 - S1 - Q2

(a) $f_1 = ?$

Solution

$$\begin{bmatrix} i \\ f \\ o \\ g \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$C_t = f \odot C_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(C_t)$$

$$f_1 = \sigma\left[W_f \begin{pmatrix} h_0 \\ x_1 \end{pmatrix} \right]$$

$$= \sigma\left[\begin{bmatrix} 0.1 & 0.2 & 0.5 & 0.6 \\ 0.3 & 0.4 & 0.7 & 0.8 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 2 \\ 1 \end{bmatrix} \right]$$

$$= \sigma \begin{bmatrix} 1 + 0.6 \\ 1.4 + 0.8 \end{bmatrix}$$

$$= \sigma \begin{bmatrix} 1.6 \\ 2.2 \end{bmatrix} \qquad \sigma(1.6) = \frac{1}{1+e^{-1.6}} \approx 0.8320$$

$$= \begin{bmatrix} 0.832 \\ 0.900 \end{bmatrix}$$

comment: The cell state at $t=1$ retains most of the memory from the previous cell state $t=0$

(ii) $C_1 = ?$

Solution

$$C_t = f \odot C_{t-1} + i \odot g$$

$$C_1 = f_1 \odot C_0 + i_1 \odot g_1$$

$$= \begin{bmatrix} 0.832 \\ 0.900 \end{bmatrix} \odot \begin{bmatrix} 0.1 \\ 0.2 \end{bmatrix} + \begin{bmatrix} 0.3 \\ 0.4 \end{bmatrix} \odot \begin{bmatrix} 0.5 \\ 0.6 \end{bmatrix}$$

$$= \begin{bmatrix} 0.0832 \\ 0.180 \end{bmatrix} + \begin{bmatrix} 0.15 \\ 0.24 \end{bmatrix}$$

$$= \begin{bmatrix} 0.233 \\ 0.420 \end{bmatrix}$$

$$\begin{array}{r} 0.15 \\ 0.0832 \\ \hline 0.2332 \end{array}$$

$$\begin{array}{r} 0.24 \\ 0.18 \\ \hline 0.42 \end{array}$$

(iii) $h_1 = ?$

Solution

$$h_t = o \odot \tanh(C_t)$$

$$h_1 = o_1 \odot \tanh(c_1)$$

$$= \begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix} \odot \tanh \begin{bmatrix} 0.233 \\ 0.420 \end{bmatrix} \quad \downarrow \text{casio}$$

$$= \begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix} \odot \begin{bmatrix} 0.229 \\ 0.397 \end{bmatrix}$$

$$= \begin{bmatrix} 0.0916 \\ 0.2382 \end{bmatrix}$$

(b) (i) transforme encoder

(ii) position embedding ViT

Solution

(i) ① Map input vector from pre-processing into context vectors using attention mechanism.

② Context vectors pass through feedforward layer to generate encoder outputs

③ Encoder outputs have better representation on than input vectors as they leverage the context information on other input token due to attention mechanism

(ii) ① since poisition embedding in ViT are crucial for incorporating spatial information about the image patches.

② Since transformer architecture lack inherent positional awareness, it treats input tokens as a set without order. Position embedding can help model understands the arrangement of patches

(c) ① Hierachical feature Representation

The hierarchy enable the model capture representations at multiple scales

② Local Self-Attention with shifted Windows

It has linear computation complexity to input image size. In contrast, vision Transformer have quadratic computation complexity to input image size due to self-attention globally

③ Scalability to High-Resolution Image.

Swin Transformer is more scalable to high-resolution due to its linear computational complexity. In contrast ViT produce feature maps of a single low resolution.