Figure 1: Illustration of the Transformer-XL model with a segment length 4.

Figure 2: Illustration of the vanilla model with a segment length 4.

Figure 3: Illustration of the relative positional encoding in the Transformer-XL model with a segment length 4.

Figure 4: Illustration of the relative positional encoding in the vanilla model with a segment length 4.

$$\mathbf{h}_{\tau+1}^{n-1} = f(\mathbf{h}_{\tau+1}^{n-2}, \mathbf{s}_{\tau+1}) \quad \text{and} \quad \mathbf{h}_{\tau}^{n-1} = f(\mathbf{h}_{\tau}^{n-2}, \mathbf{s}_{\tau}),$$

where f is a function with parameter $n-2$. Now, the last layer can use the first segment $\mathbf{h}_{\tau}^{n-1}$ to produce the next segment $\mathbf{h}_{\tau+1}^{n}$. Therefore, the maximum possible dependency length grows with respect to the number of layers, as well as the segment length.

Finally, to update the hidden states, for each segment, the hidden state sequence needs to be computed in a recursive fashion from the lowest layer to the highest layer. Therefore, the evaluation of the new segment depends on the old segment.

Segment-Level Recurrence with State Reuse

To address the limitations of using only the previous segment, we propose to also reuse the hidden states of previous segments. The idea is to cache the hidden state sequence to be reused (without gradient) when the model processes the next segment. Figure 2 illustrates the idea with a segment length 4