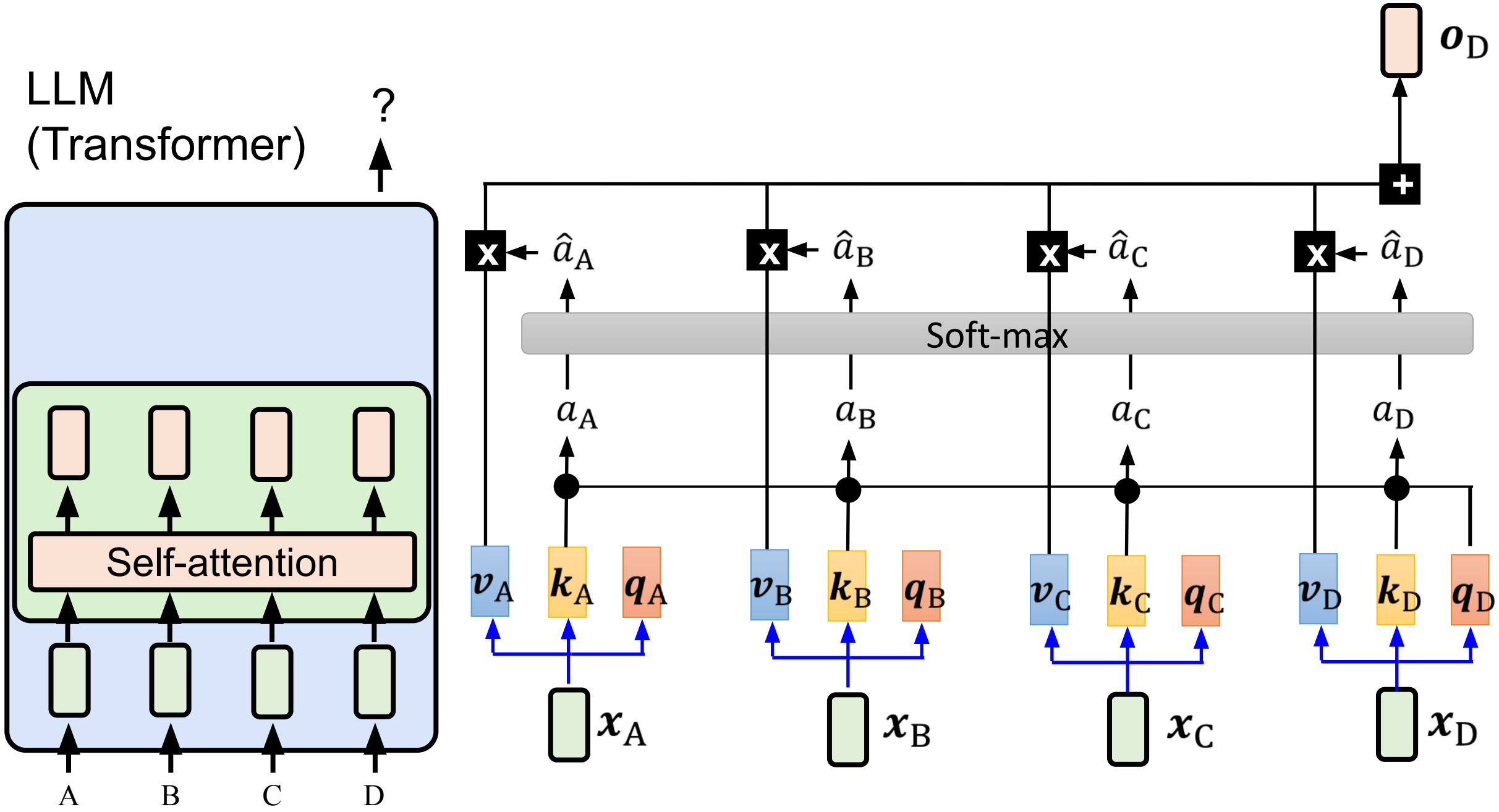
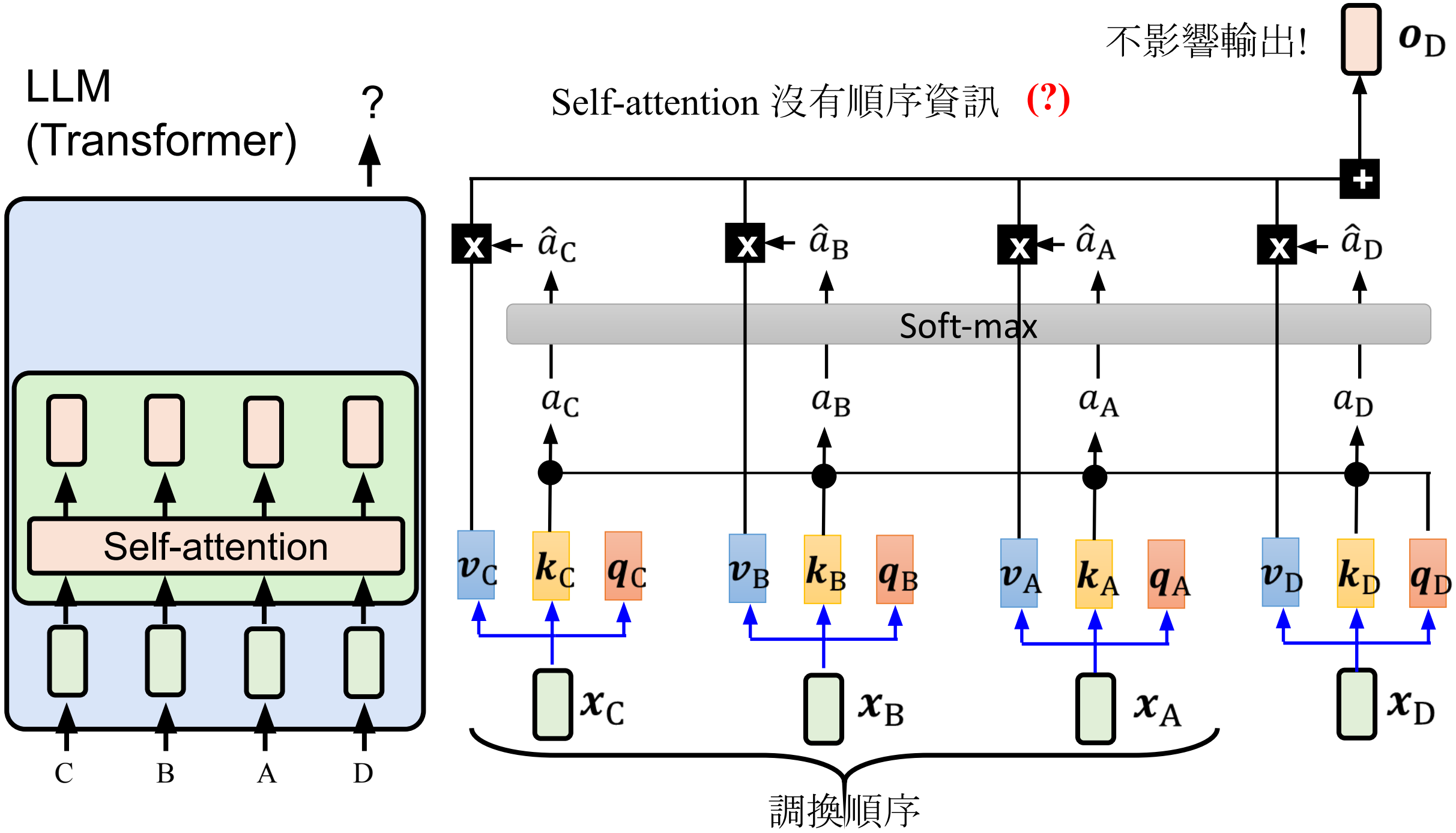
Transformer 怎麼知道 Token 的順序？

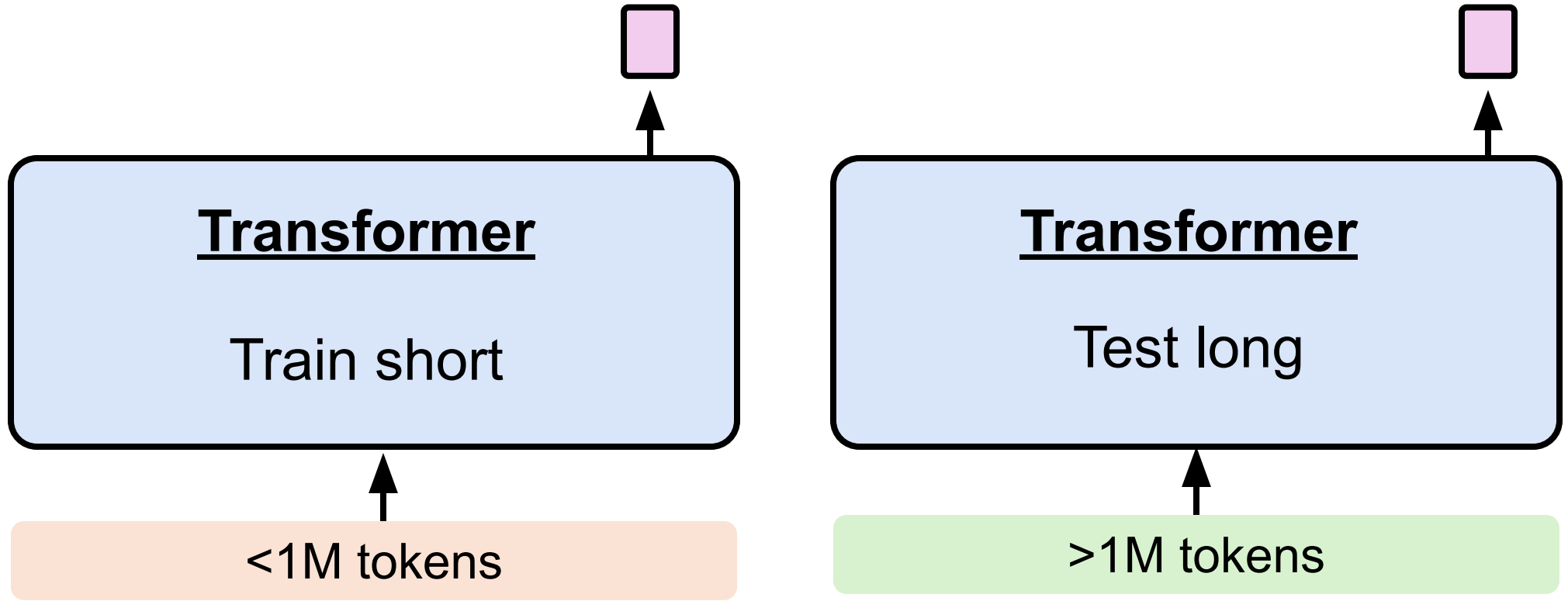
Positional Embedding


LLM
(Transformer)
?
Self-attention
A
B
C
D
o_D
$\hat{a}_A$
$\hat{a}_B$
$\hat{a}_C$
$\hat{a}_D$
Soft-max
a_A
a_B
a_C
a_D
v_A k_A q_A
v_B k_B q_B
v_C k_C q_C
v_D k_D q_D
x_A
x_B
x_C
x_D


LLM
(Transformer)
?
Self-attention
C
B
A
D
Self-attention 沒有順序資訊 (?)
不影響輸出!
o_D
$\hat{a}_C$
$\hat{a}_B$
$\hat{a}_A$
$\hat{a}_D$
Soft-max
a_C
a_B
a_A
a_D
v_C k_C q_C
v_B k_B q_B
v_A k_A q_A
v_D k_D q_D
x_C
x_B
x_A
x_D
調換順序

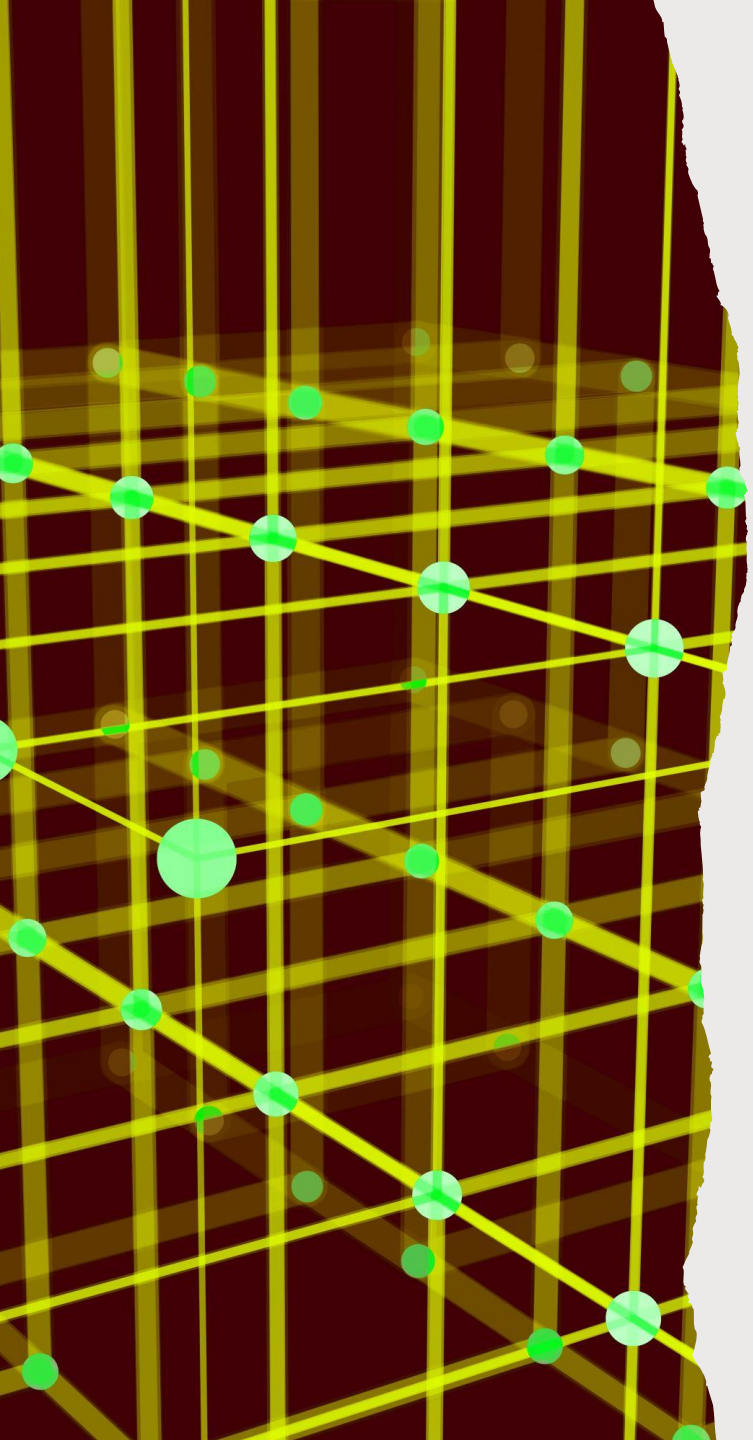
Zero-shot Long Context

- Train short, test long



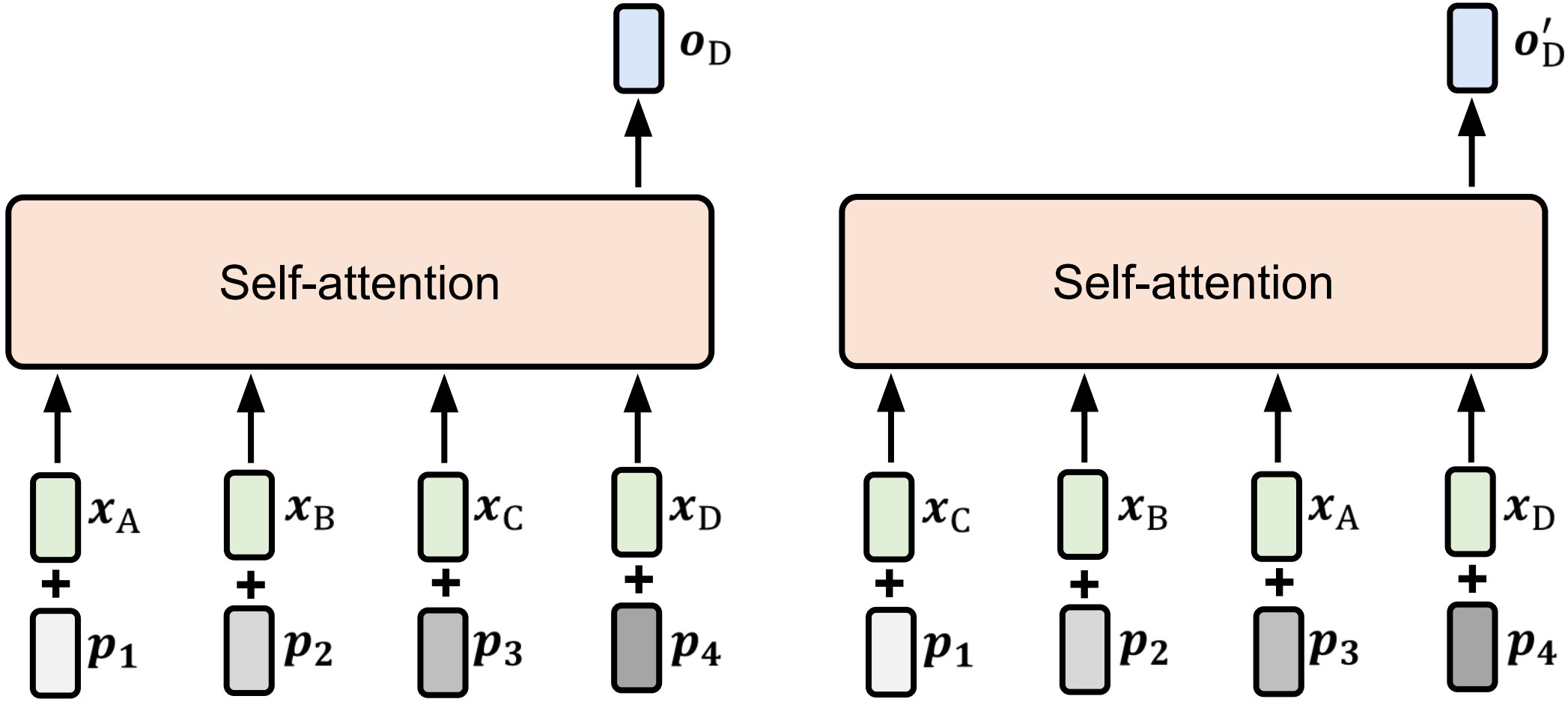
Outline

- Absolute Positional Embedding
- Relative Positional Embedding
- RoPE
- Train short, test long
- No Positional Embedding?!

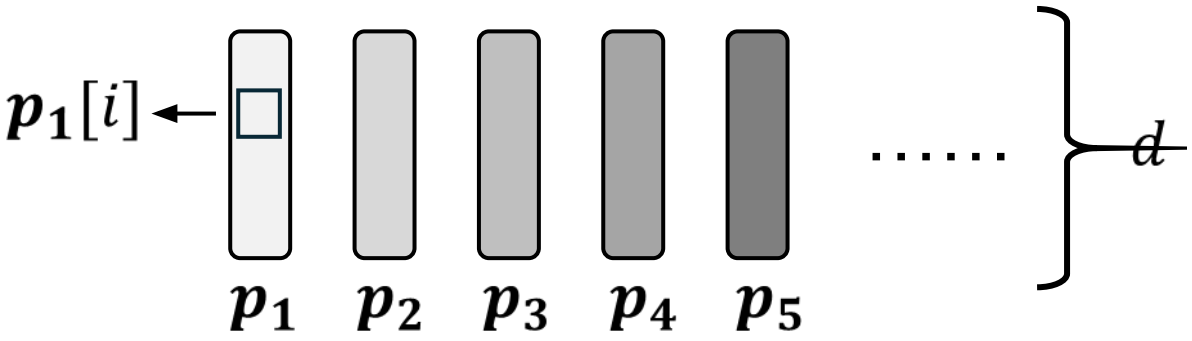


Absolute positional embedding

Absolute Positional Embedding

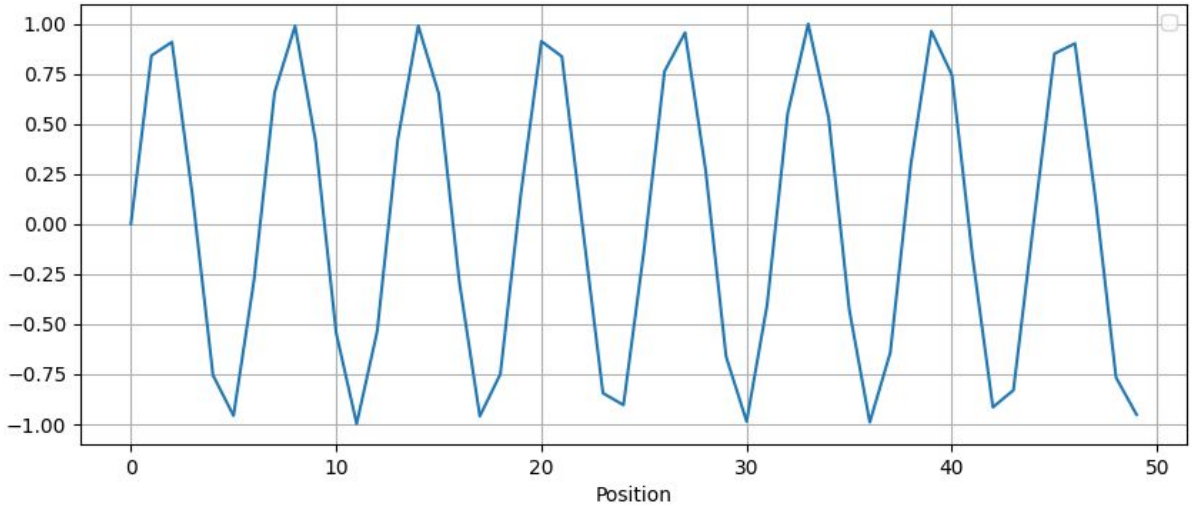


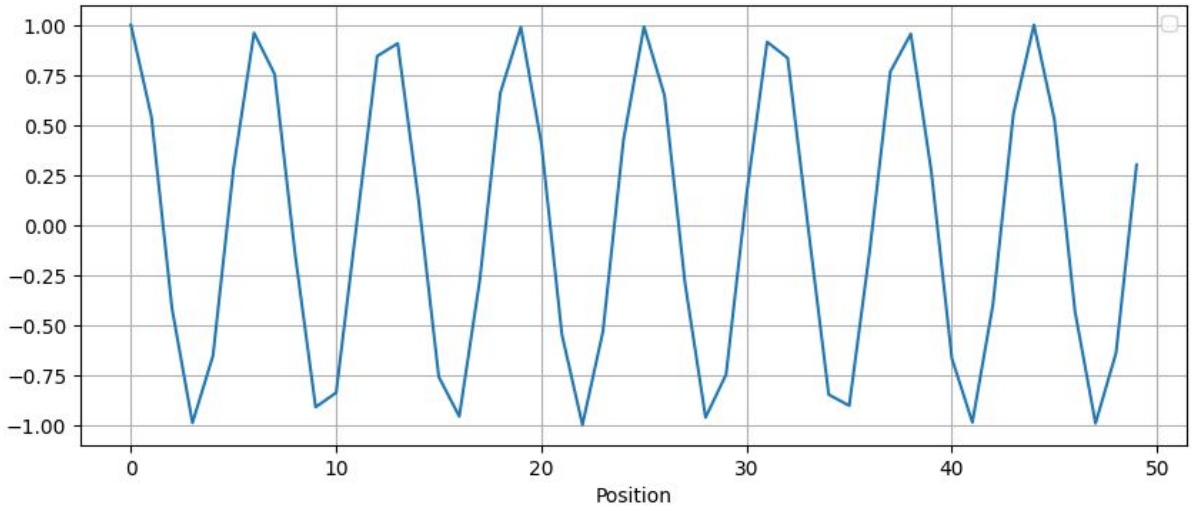
Sinusoidal Positional Embedding



$$p_k[2i] = sin\left(\frac{k}{10000^{2i/d}}\right)$$

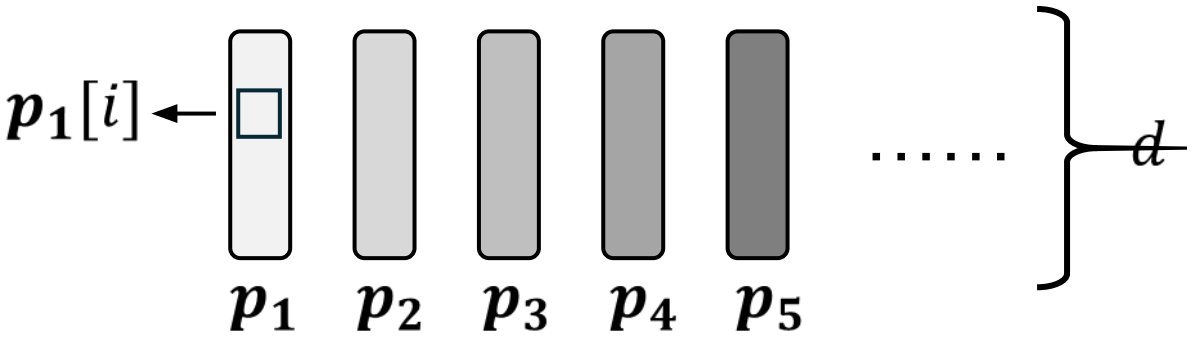
$$p_k[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)$$





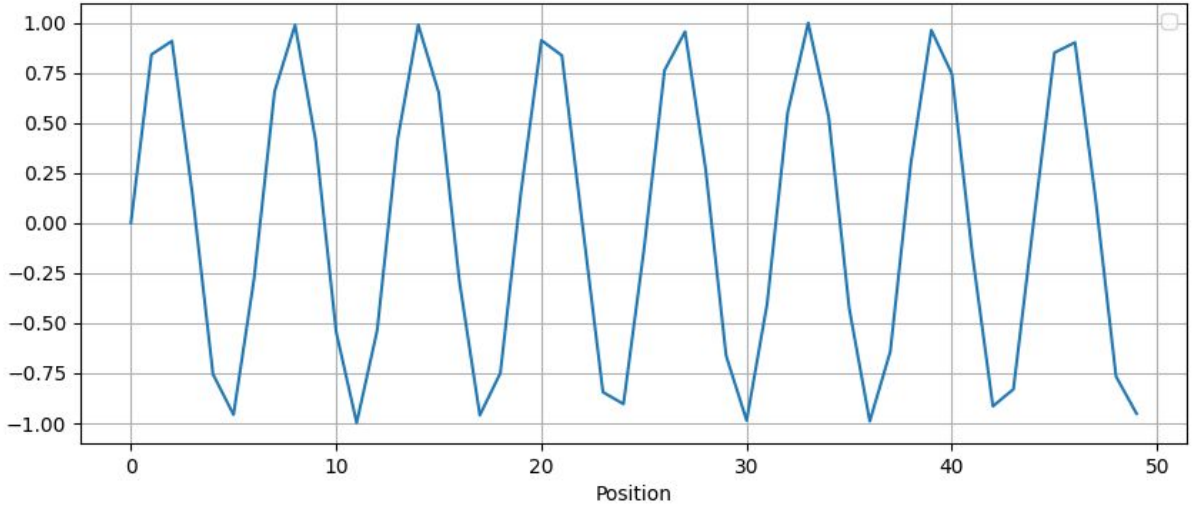
https://arxiv.org/abs/1706.03762

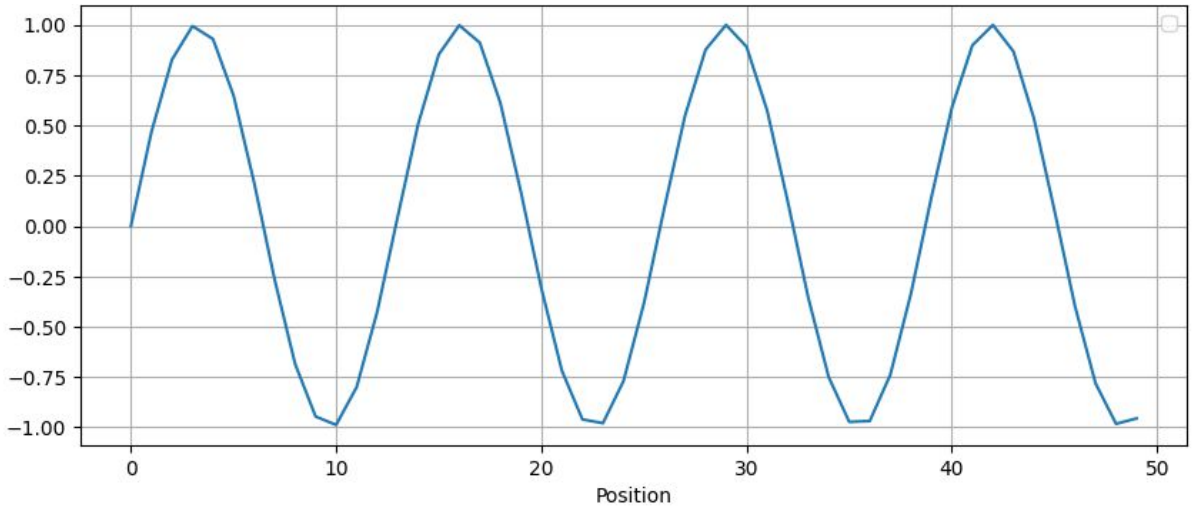
Sinusoidal Positional Embedding



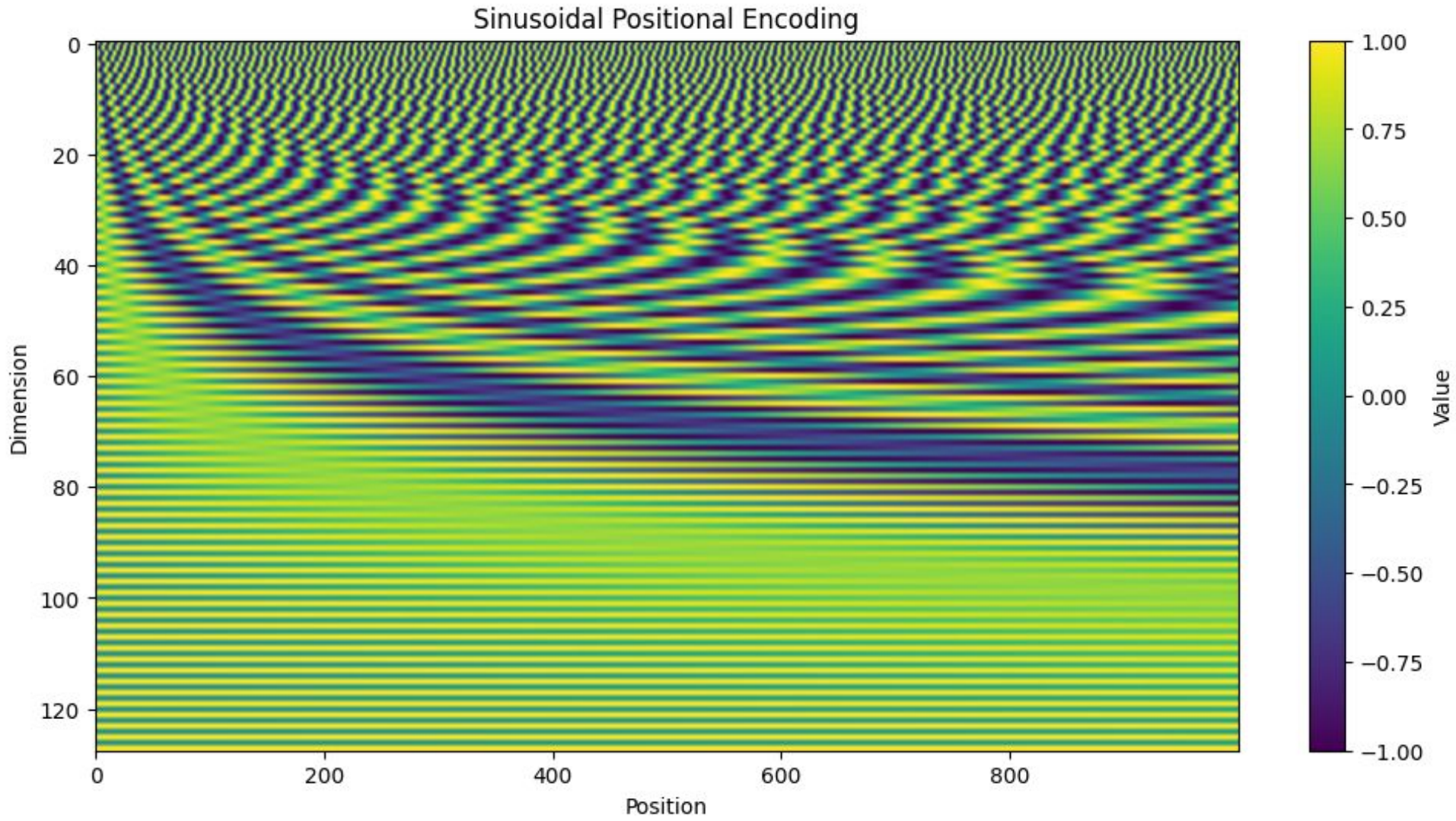
$$p_k[2i] = sin\left(\frac{k}{10000^{2i/d}}\right)$$

$$p_k[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)$$





Sinusoidal Positional Embedding



Sinusoidal Positional Embedding

$$\boldsymbol{p_k}[2i] = sin\left(\frac{k}{10000^{2i/d}}\right) \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)$$

$$\frac{k}{10000^{2i/d}} = 2\pi$$

$$k = 2\pi \cdot 10000^{2i/d}$$

$$d = 128$$

$i = 0$ (0,1 dim) `6.3`

$i = 32$ (64,65 dim) `628.3`

$i = 63$ (126,127 dim) `54410.1`

3.5 Positional Encoding

Since our model contains no recurrence and no convolution, in order for the model to make use of the order of the sequence, we must inject some information about the relative or absolute position of the tokens in the sequence. To this end, we add "positional encodings" to the input embeddings at the bottoms of the encoder and decoder stacks. The positional encodings have the same dimension d_{model} as the embeddings, so that the two can be summed. There are many choices of positional encodings, learned and fixed [9].

In this work, we use sine and cosine functions of different frequencies:

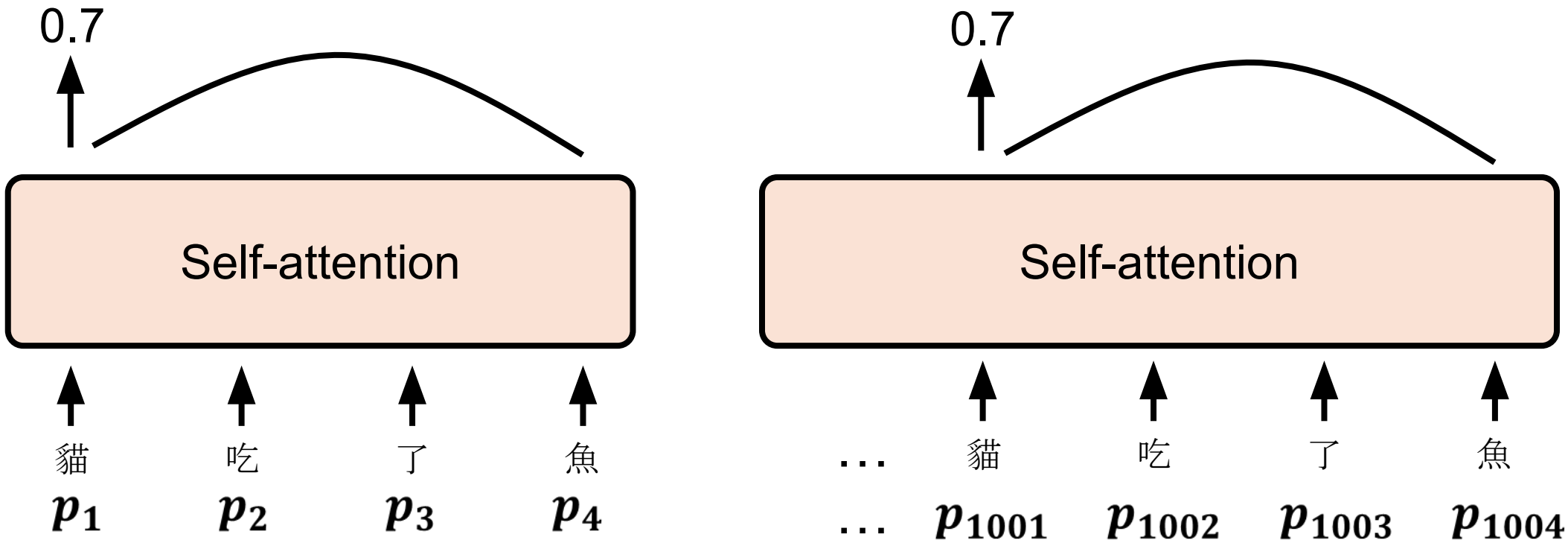
$$PE_{(pos,2i)} = sin(pos/10000^{2i/d_{\text{model}}})$$
$$PE_{(pos,2i+1)} = cos(pos/10000^{2i/d_{\text{model}}})$$

where pos is the position and i is the dimension. That is, each dimension of the positional encoding corresponds to a sinusoid. The wavelengths form a geometric progression from 2π to $10000 \cdot 2\pi$. We chose this function because we hypothesized it would allow the model to easily learn to attend by relative positions, since for any fixed offset k, PE_{pos+k} can be represented as a linear function of PE_{pos}.

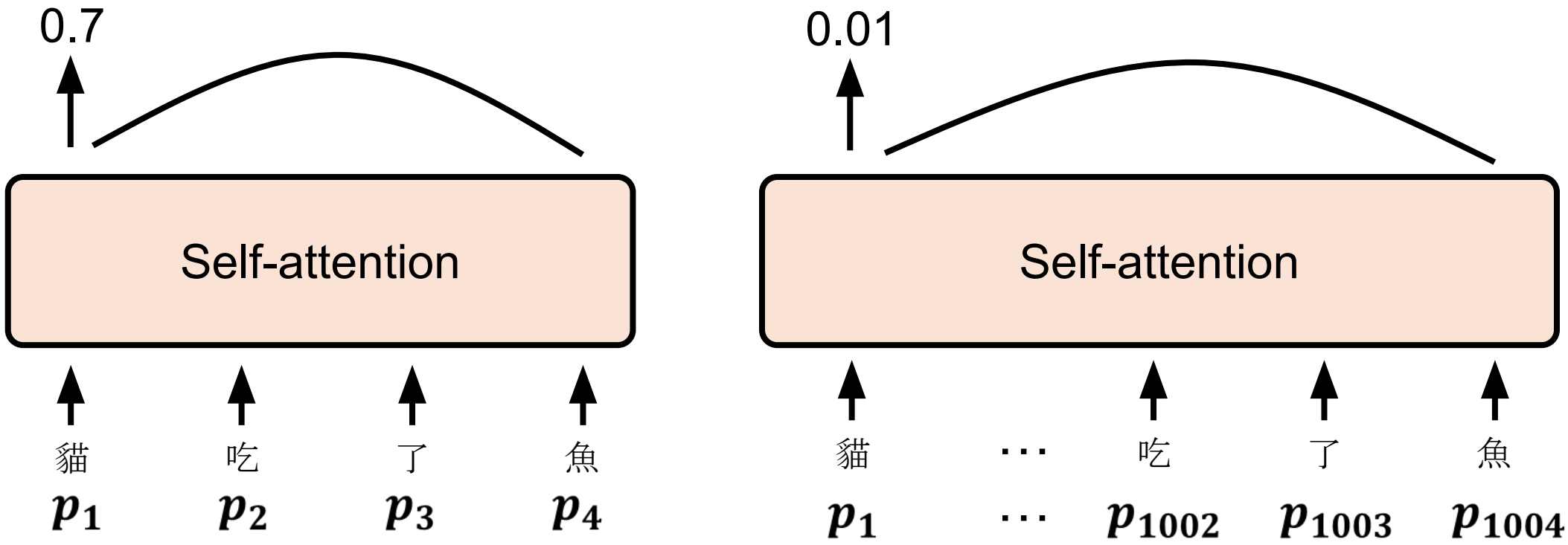
We also experimented with using learned positional embeddings [9] instead, and found that the two versions produced nearly identical results (see Table 3 row (E)). We chose the sinusoidal version because it may allow the model to extrapolate to sequence lengths longer than the ones encountered during training.

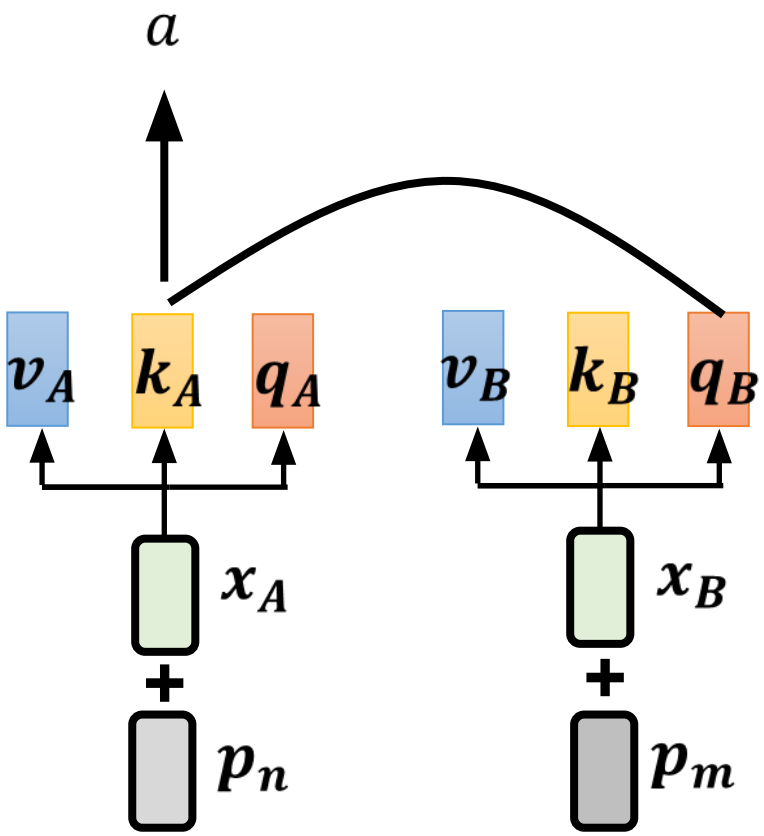
https://arxiv.org/abs/1706.03762

Relative positioning is crucial!



Relative positioning is crucial!





$$a = \boldsymbol{q_B} \cdot \boldsymbol{k_A} \quad = (\boldsymbol{q_B})^T \boldsymbol{k_A}$$

$$= \left(W_q(\boldsymbol{x_B} + \boldsymbol{p_m})\right)^T W_k(\boldsymbol{x_A} + \boldsymbol{p_n})$$

$$= (\boldsymbol{x_B} + \boldsymbol{p_m})^T {W_q}^T W_k(\boldsymbol{x_A} + \boldsymbol{p_n})$$

$$= \boldsymbol{x_B}^T {W_q}^T W_k \boldsymbol{x_A}$$ 只跟內容有關

$$+\boldsymbol{x}_\mathrm{B}^T {W_q}^T W_k \boldsymbol{p_n} + \boldsymbol{p_m}^T {W_q}^T W_k \boldsymbol{x_A}$$ 內容、位置交互影響

$$+\boldsymbol{p_m}^T {W_q}^T W_k \boldsymbol{p_n}$$ 只跟位置有關

$m = 3, n = 1$ 和 $m = 1003, n = 1001$
沒有甚麼關聯

Back to Sinusoidal Positional Embedding

$$\boldsymbol{p_{k+r}} = M_r \boldsymbol{p_k}$$

$$\boldsymbol{p_k}[2i] = sin\left(\frac{k}{10000^{2i/d}}\right)_z \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)_z$$

$$\boldsymbol{p_{k+r}}[2i] = sin\left(\frac{k+r}{z}\right) \qquad \boldsymbol{p_k}[2i] = sin\left(\frac{k}{z}\right)$$

$$\boldsymbol{p_{k+r}}[2i+1] = cos\left(\frac{k+r}{z}\right) \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{z}\right)$$

$$sin(a+b) = sin(a)cos(b) + cos(a)sin(b)$$

$$cos(a+b) = cos(a)cos(b) - sin(a)sin(b)$$

Back to Sinusoidal Positional Embedding

$$\boldsymbol{p_{k+r}} = M_r \boldsymbol{p_k} \qquad \boldsymbol{p_k}[2i] = sin\left(\frac{k}{10000^{2i/d}}\right) \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)$$

z z

$$\boldsymbol{p_{k+r}}[2i] = sin\left(\frac{k+r}{z}\right) = sin\left(\frac{k}{z}\right)cos\left(\frac{r}{z}\right) + cos\left(\frac{k}{z}\right)sin\left(\frac{r}{z}\right)$$

$$\boldsymbol{p_{k+r}}[2i+1] = cos\left(\frac{k+r}{z}\right) = cos\left(\frac{k}{z}\right)cos\left(\frac{r}{z}\right) - sin\left(\frac{k}{z}\right)sin\left(\frac{r}{z}\right)$$

$$\boldsymbol{p_k}[2i] = sin\left(\frac{k}{z}\right)$$

$$\boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{z}\right)$$

$$sin(a+b) = sin(a)cos(b) + cos(a)sin(b)$$

$$cos(a+b) = cos(a)cos(b) - sin(a)sin(b)$$

$$\boldsymbol{p_{k+r}} = M_r \boldsymbol{p_k} \qquad \boldsymbol{p_k}[2i] = sin\left(\frac{k}{10000^{2i/d}}\right)_z \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)_z$$

$$\boldsymbol{p_{k+r}}[2i] = sin\left(\frac{k+r}{z}\right) = sin\left(\frac{k}{z}\right)cos\left(\frac{r}{z}\right) + cos\left(\frac{k}{z}\right)sin\left(\frac{r}{z}\right)$$

$$= \boldsymbol{p_k}[2i]cos\left(\frac{r}{z}\right) + \boldsymbol{p_k}[2i+1]sin\left(\frac{r}{z}\right)$$

$$\boldsymbol{p_{k+r}}[2i+1] = cos\left(\frac{k+r}{z}\right) = cos\left(\frac{k}{z}\right)cos\left(\frac{r}{z}\right) - sin\left(\frac{k}{z}\right)sin\left(\frac{r}{z}\right)$$

$$\boldsymbol{p_k}[2i] = sin\left(\frac{k}{z}\right) \qquad = \boldsymbol{p_k}[2i+1]cos\left(\frac{r}{z}\right) - \boldsymbol{p_k}[2i]sin\left(\frac{r}{z}\right)$$

$$\boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{z}\right)$$

$$\boldsymbol{p_{k+r}} = M_r \boldsymbol{p_k} \qquad \boldsymbol{p_k}[2i] = sin\left(\frac{k}{10000^{2i/d}}\right) \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)$$

z z

$$\boldsymbol{p_{k+r}}[2i] = \boldsymbol{p_k}[2i]cos\left(\frac{r}{z}\right) + \boldsymbol{p_k}[2i+1]sin\left(\frac{r}{z}\right)$$

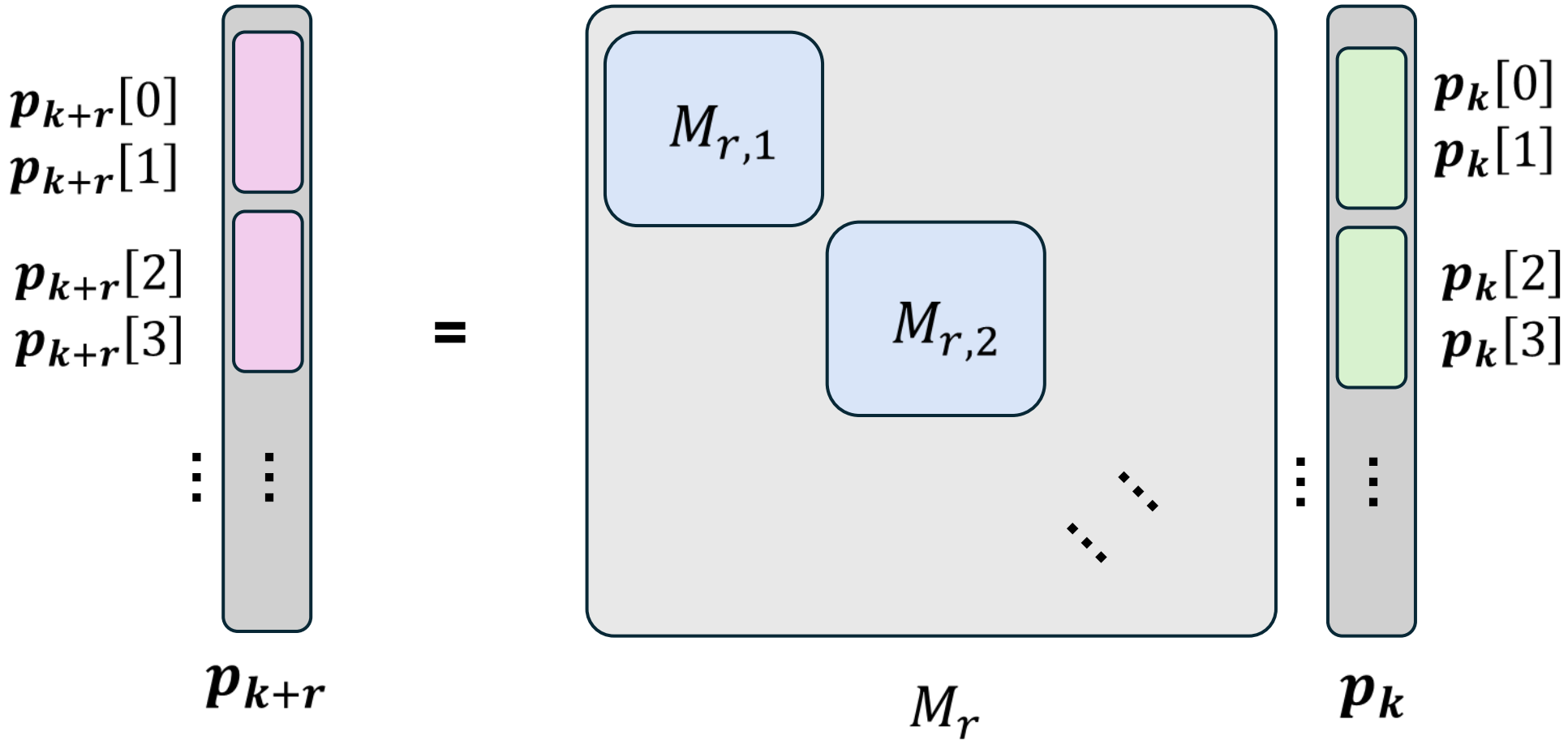
$$\boldsymbol{p_{k+r}}[2i+1] = \boldsymbol{p_k}[2i+1]cos\left(\frac{r}{z}\right) - \boldsymbol{p_k}[2i]sin\left(\frac{r}{z}\right)$$

$$\begin{bmatrix} \boldsymbol{p_{k+r}}[2i] \\ \boldsymbol{p_{k+r}}[2i+1] \end{bmatrix} = \begin{bmatrix} cos\left(\frac{r}{z}\right) & sin\left(\frac{r}{z}\right) \\ -sin\left(\frac{r}{z}\right) & cos\left(\frac{r}{z}\right) \end{bmatrix} \begin{bmatrix} \boldsymbol{p_k}[2i] \\ \boldsymbol{p_k}[2i+1] \end{bmatrix}$$

$M_{r,i}$

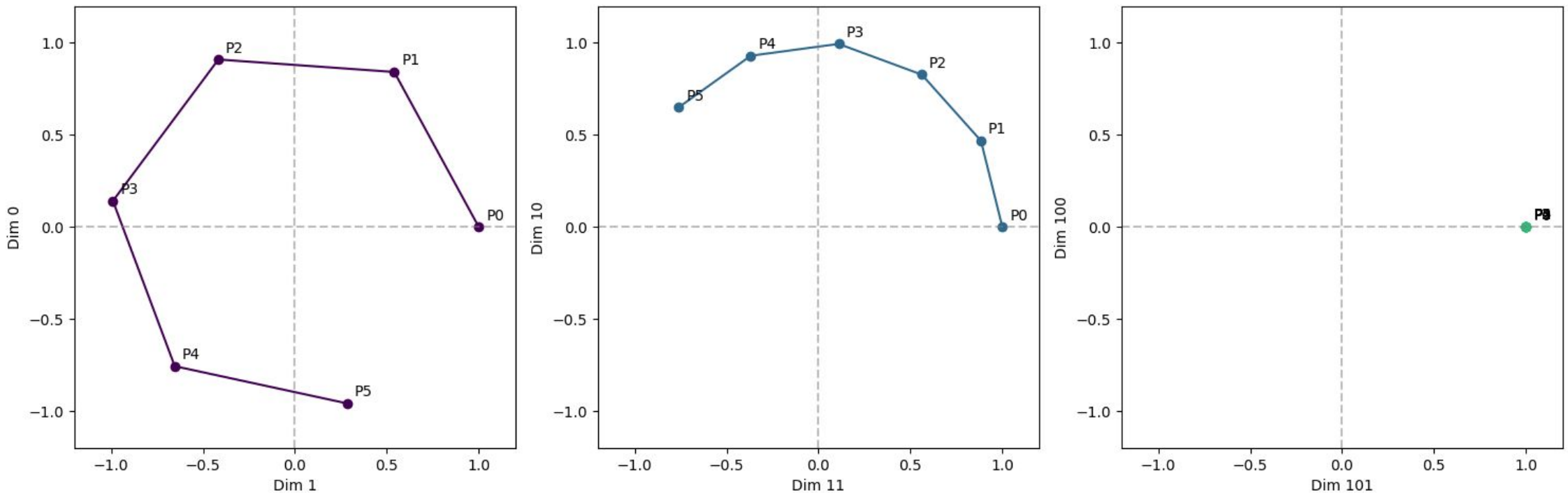
$$\boldsymbol{p_{k+r}} = M_r \boldsymbol{p_k} \qquad \boldsymbol{p_k}[2i] = sin\left(\frac{k}{10000^{2i/d}}\right) \qquad \boldsymbol{p_k}[2i+1] = cos\left(\frac{k}{10000^{2i/d}}\right)$$

$$\begin{bmatrix} \boldsymbol{p_{k+r}}[2i] \\ \boldsymbol{p_{k+r}}[2i+1] \end{bmatrix} = M_{r,i} \begin{bmatrix} \boldsymbol{p_k}[2i] \\ \boldsymbol{p_k}[2i+1] \end{bmatrix}$$

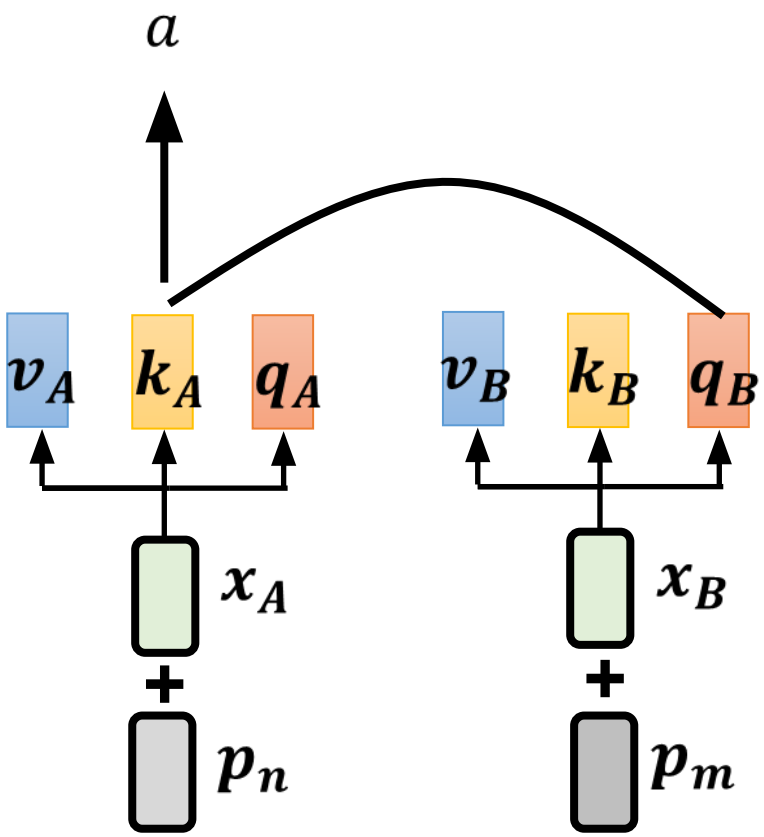


$$\begin{bmatrix} \boldsymbol{p_{k+r}}[2i] \\ \boldsymbol{p_{k+r}}[2i+1] \end{bmatrix} = \begin{bmatrix} cos\left(\frac{r}{z}\right) & sin\left(\frac{r}{z}\right) \\ -sin\left(\frac{r}{z}\right) & cos\left(\frac{r}{z}\right) \end{bmatrix} \begin{bmatrix} \boldsymbol{p_k}[2i] \\ \boldsymbol{p_k}[2i+1] \end{bmatrix}$$

$$M_{r,i}$$



$$\boldsymbol{p}_{k+r} = M_r \boldsymbol{p}_k$$



$$a = \boldsymbol{q}_B \cdot \boldsymbol{k}_A \quad = (\boldsymbol{q}_B)^T \boldsymbol{k}_A$$

$$= \left(W_q(\boldsymbol{x}_B + \boldsymbol{p}_m)\right)^T W_k(\boldsymbol{x}_A + \boldsymbol{p}_n)$$

$$= (\boldsymbol{x}_B + \boldsymbol{p}_m)^T W_q{}^T W_k(\boldsymbol{x}_A + \boldsymbol{p}_n)$$

$$= \boldsymbol{x}_B{}^T W_q{}^T W_k \boldsymbol{x}_A \quad \text{只跟內容有關}$$

$$+\boldsymbol{x}_B{}^T W_q{}^T W_k \boldsymbol{p}_n + \boldsymbol{p}_m{}^T W_q{}^T W_k \boldsymbol{x}_A \quad \text{內容、位置交互影響}$$

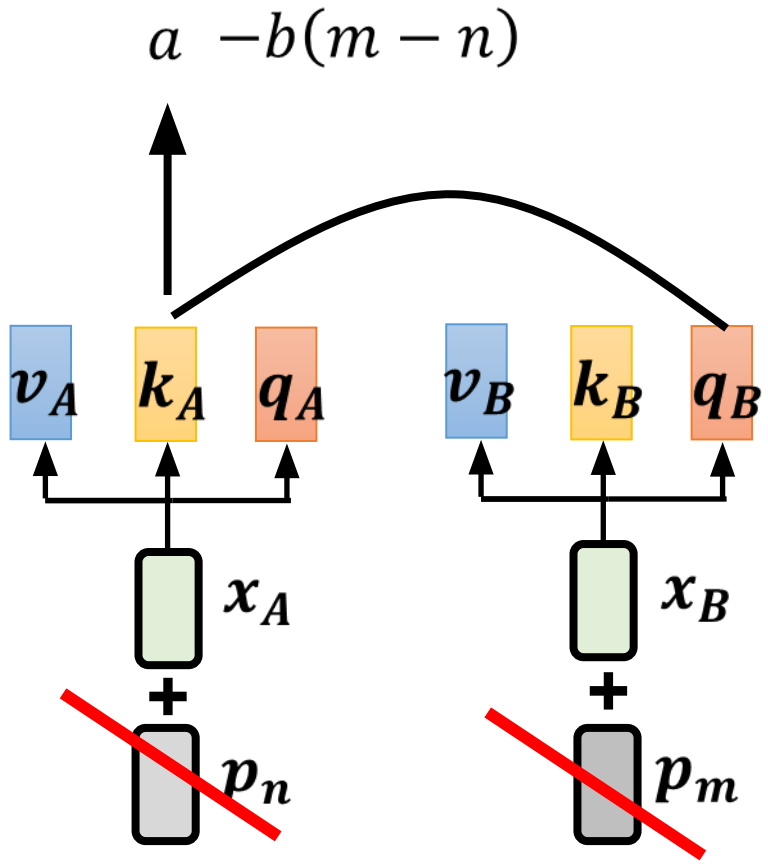
$$+\boldsymbol{p}_m{}^T W_q{}^T W_k \boldsymbol{p}_n \quad \text{只跟位置有關}$$

$$= (M_{m-n}\boldsymbol{p}_n)^T W_q{}^T W_k \boldsymbol{p}_n \quad = (\boldsymbol{p}_n)^T M_{m-n} W_q{}^T W_k \boldsymbol{p}_n$$

Relative Positional Embedding

Attention with Linear Biases (ALiBi)

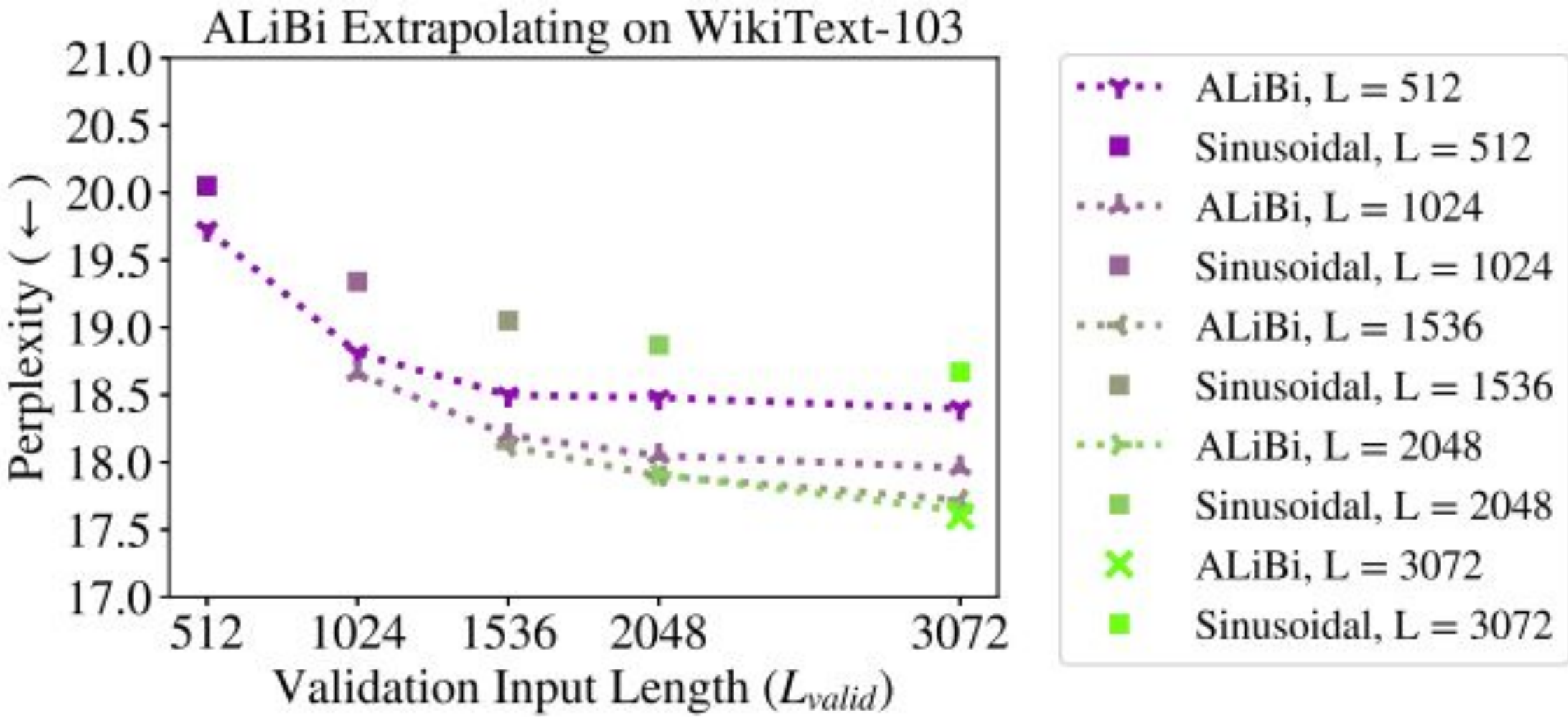
https://arxiv.org/abs/2108.12409



距離越遠，attention 越小，
就這樣!

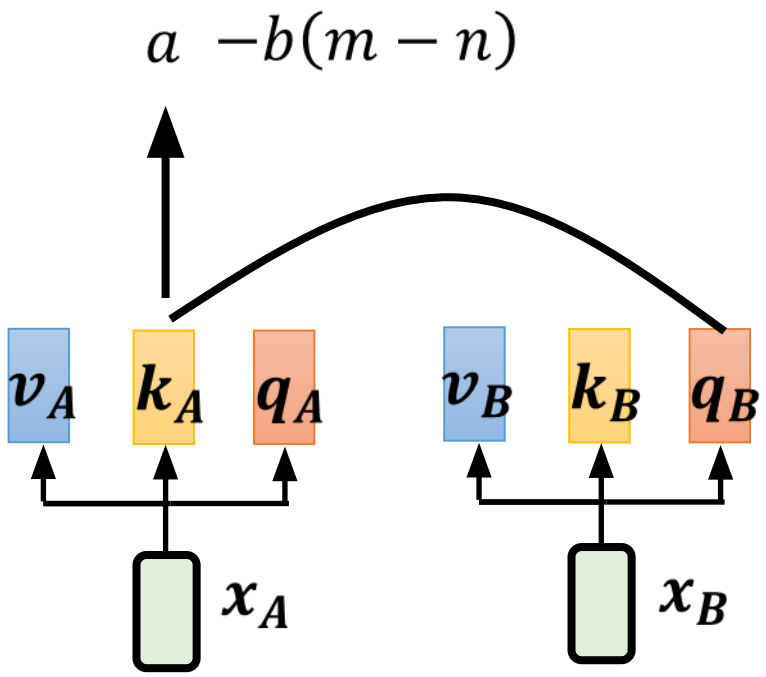
b 是手動設置

不同 attention head
設置不同值

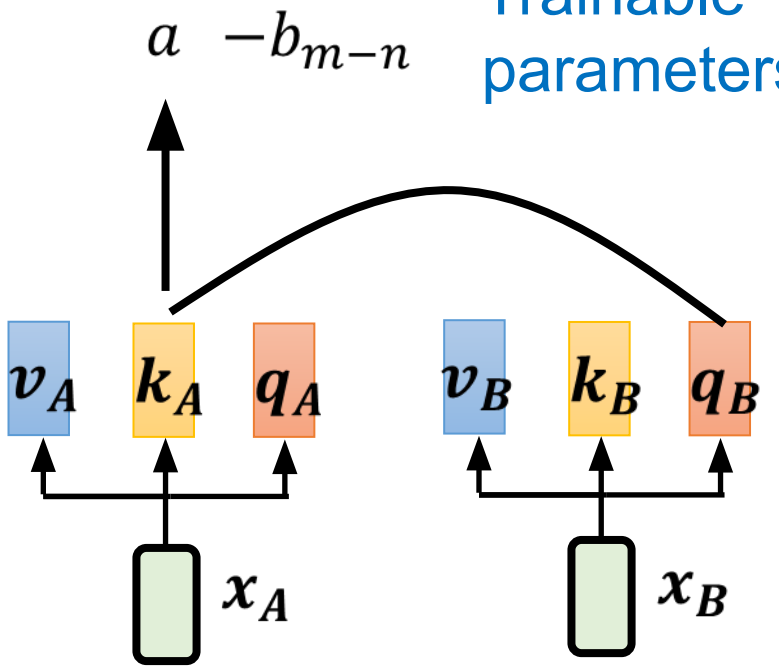


https://arxiv.org/abs/2108.12409

Text-to-Text Transfer Transformer (T5)



https://arxiv.org/abs/2108.12409



https://arxiv.org/abs/1910.10683

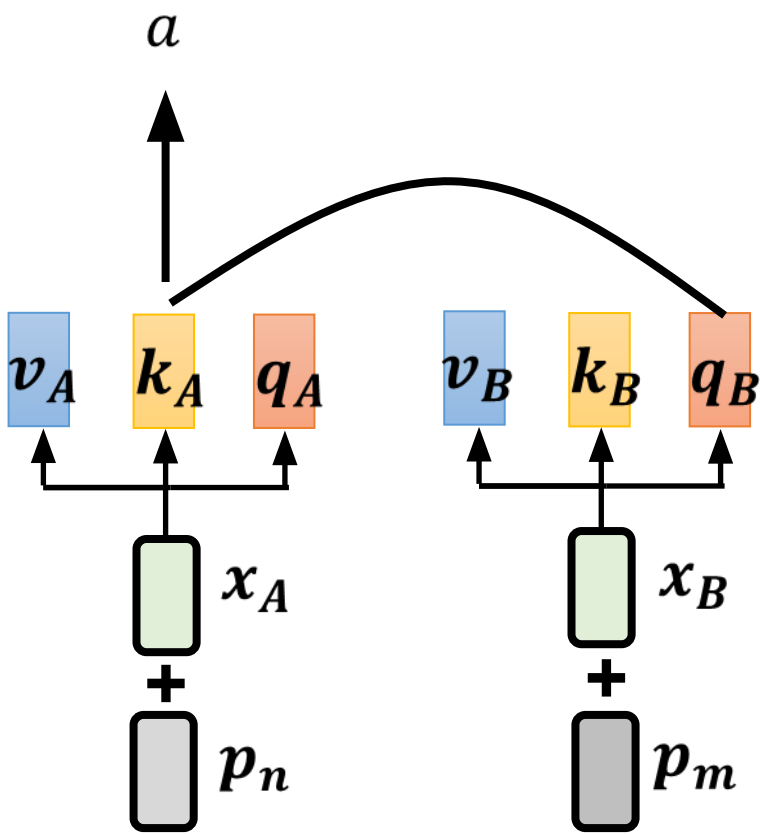
RoPE

Rotary Position Embedding

Llama、Qwen、Gemma

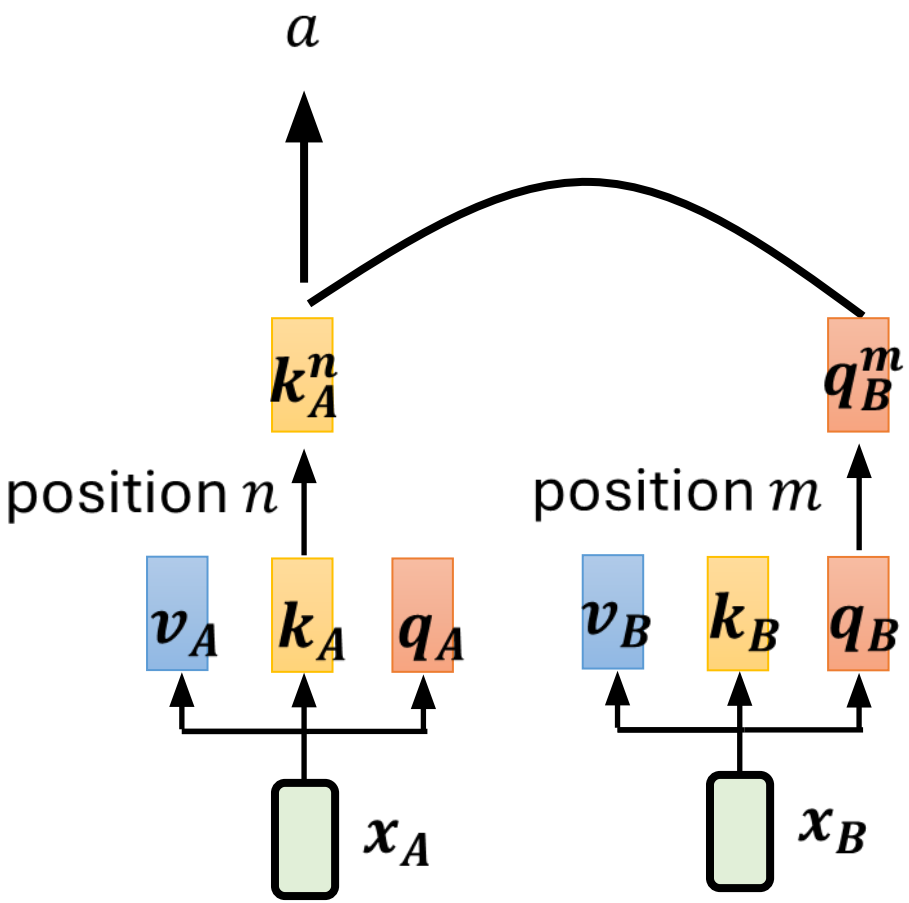
https://arxiv.org/abs/2104.09864

$$a = \left(W_q(\boldsymbol{x_A} + \boldsymbol{p_m})\right)^T W_k(\boldsymbol{x_B} + \boldsymbol{p_n})$$

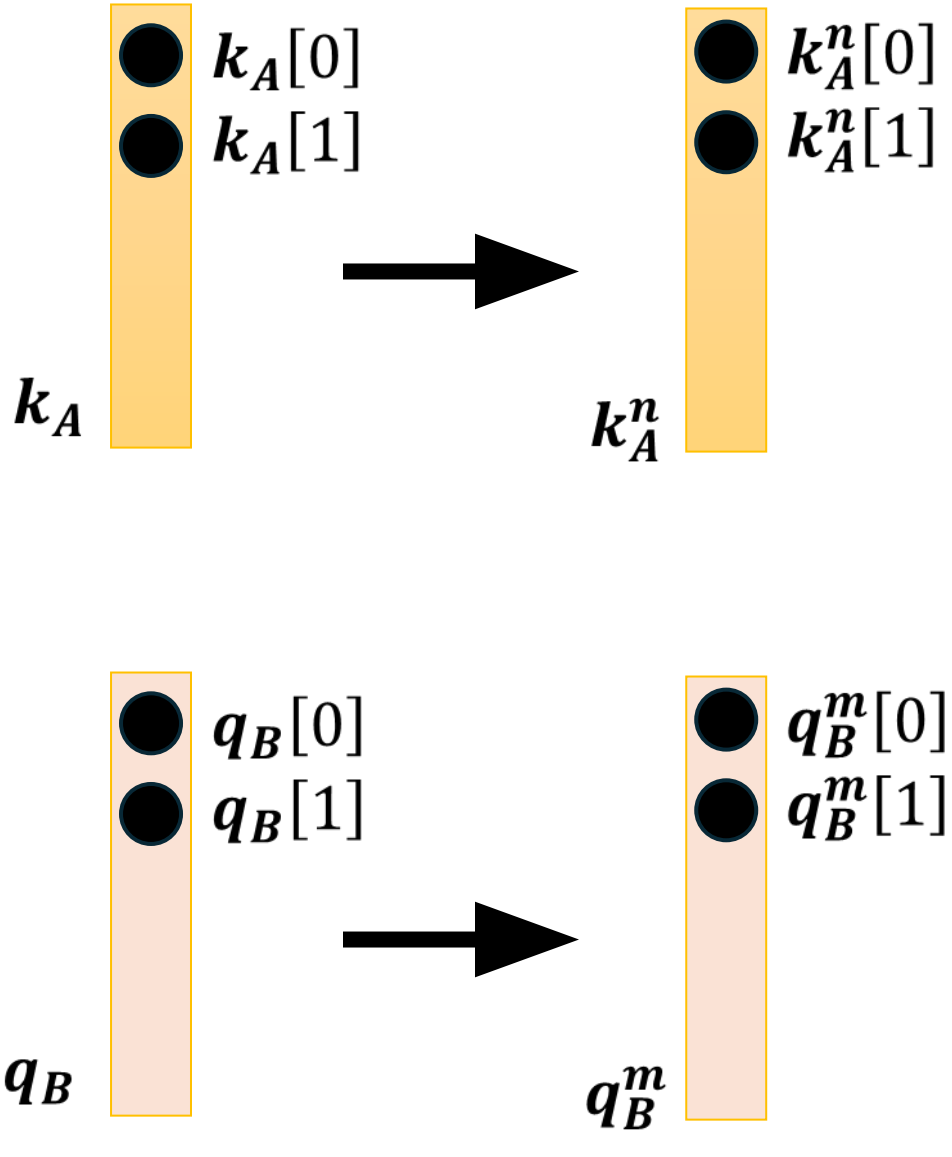


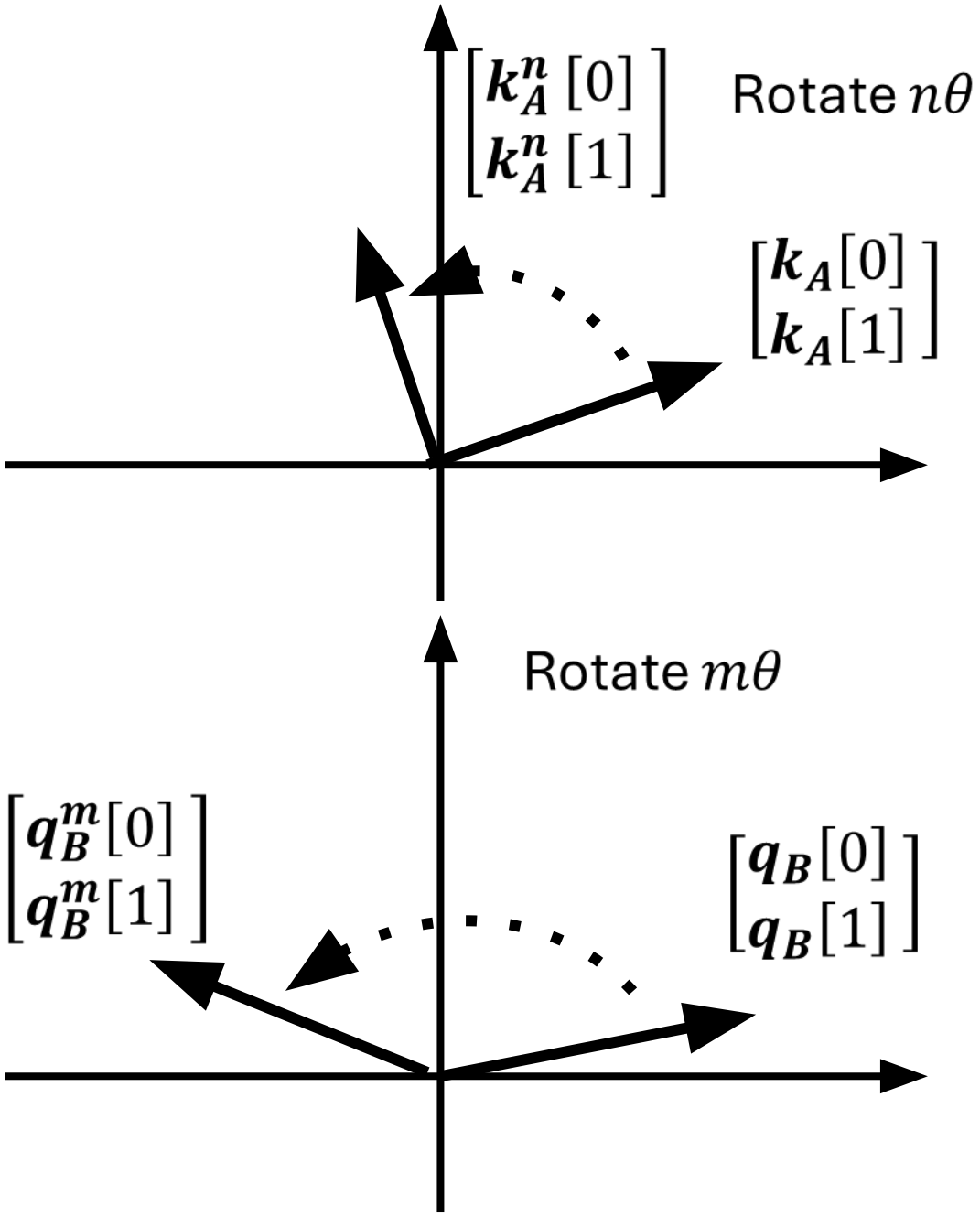
$$a = (\boldsymbol{k_A^n})^T \boldsymbol{q_B^m}$$

$$= (\boldsymbol{q_A})^T R_{m-n} \boldsymbol{q_B}$$

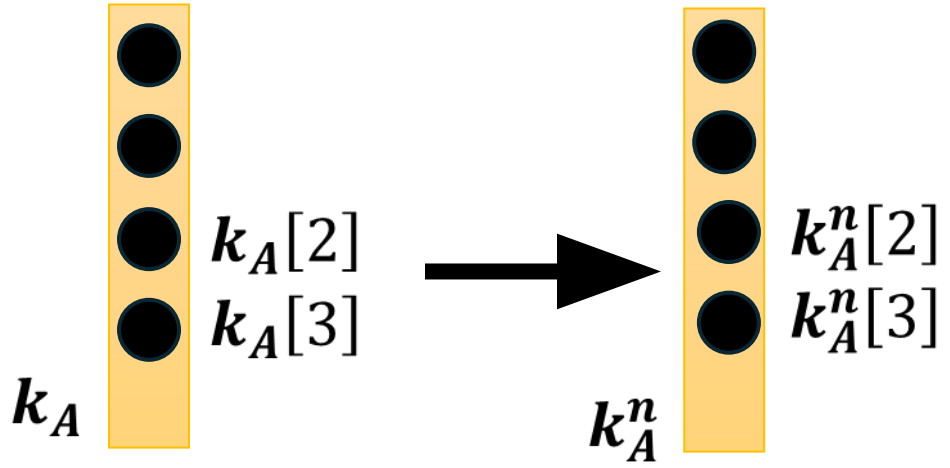


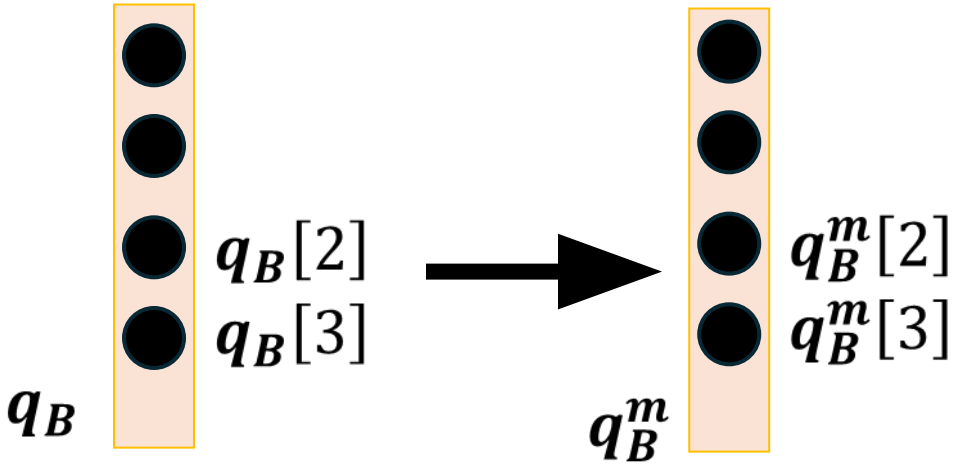
Rotation as Position

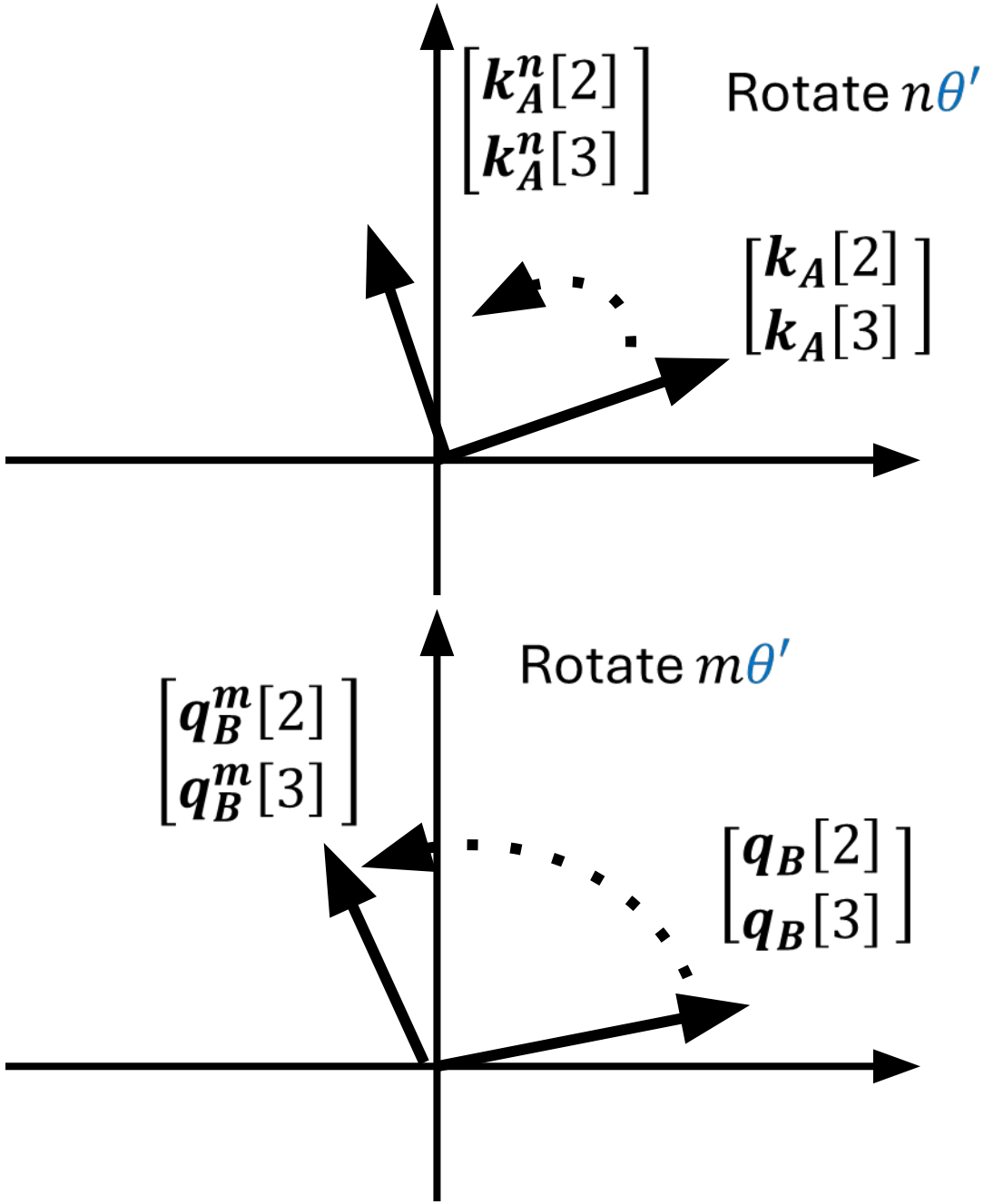




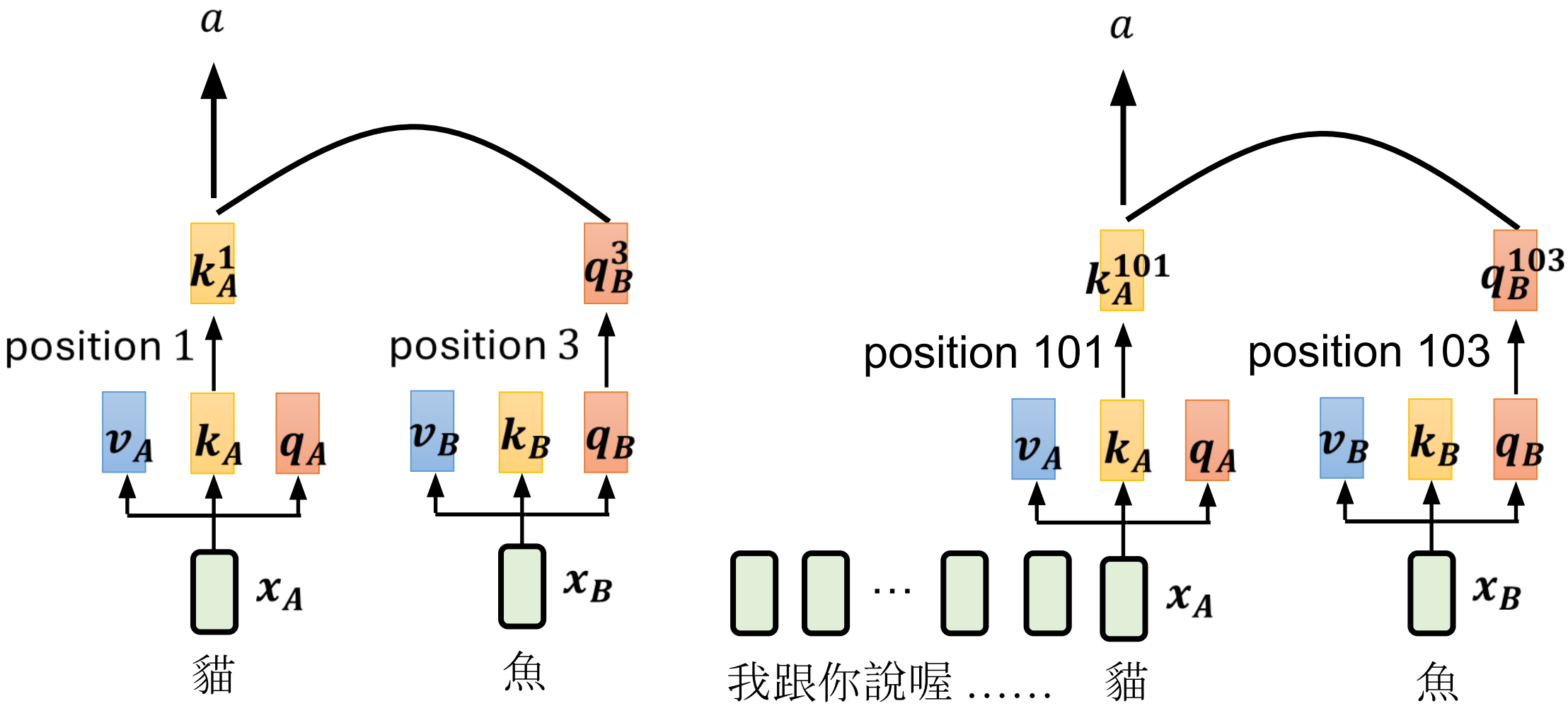
Rotation as Position

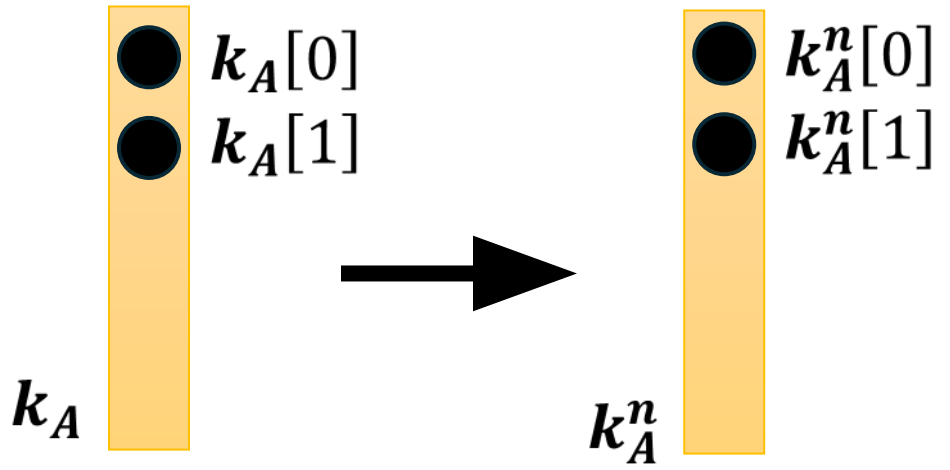






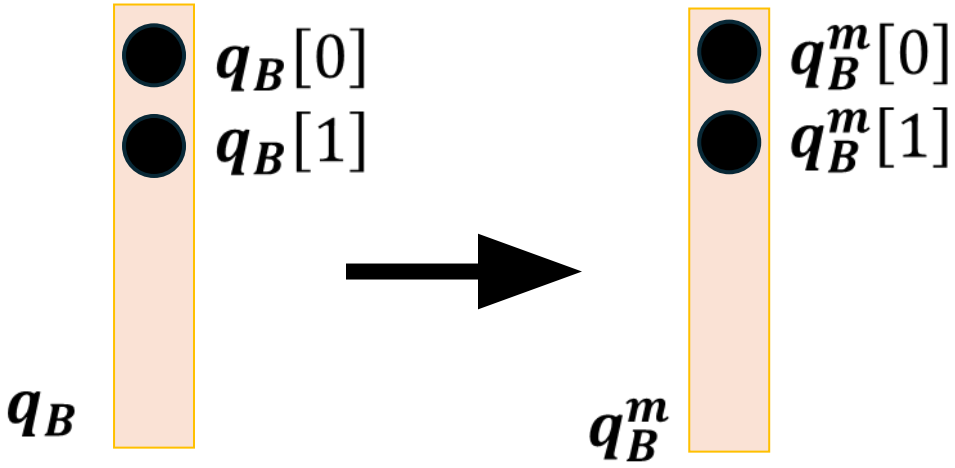
$$k_A^n \cdot q_B^m = k_A^{n+r} \cdot q_B^{m+r}$$





Rotate $n\theta$

$$\begin{bmatrix} \boldsymbol{k_A^n}[0] \\ \boldsymbol{k_A^n}[1] \end{bmatrix} = \begin{bmatrix} cos(n\theta) & -sin(n\theta) \\ sin(n\theta) & cos(n\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}$$



Rotate $m\theta$

$$\begin{bmatrix} \boldsymbol{q_B^m}[0] \\ \boldsymbol{q_B^m}[1] \end{bmatrix} = \begin{bmatrix} cos(m\theta) & -sin(m\theta) \\ sin(m\theta) & cos(m\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

Rotate $n\theta$

$$\begin{bmatrix} \boldsymbol{k_A^n}[0] \\ \boldsymbol{k_A^n}[1] \end{bmatrix} = \begin{bmatrix} cos(n\theta) & -sin(n\theta) \\ sin(n\theta) & cos(n\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}$$

Rotate $m\theta$

$$\begin{bmatrix} \boldsymbol{q_B^m}[0] \\ \boldsymbol{q_B^m}[1] \end{bmatrix} = \begin{bmatrix} cos(m\theta) & -sin(m\theta) \\ sin(m\theta) & cos(m\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{k_A^n}[0] \\ \boldsymbol{k_A^n}[1] \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{q_B^m}[0] \\ \boldsymbol{q_B^m}[1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{k_A^n}[0] \\ \boldsymbol{k_A^n}[1] \end{bmatrix}^T \begin{bmatrix} \boldsymbol{q_B^m}[0] \\ \boldsymbol{q_B^m}[1] \end{bmatrix}$$

$$= \left(\begin{bmatrix} cos(n\theta) & -sin(n\theta) \\ sin(n\theta) & cos(n\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix} \right)^T \begin{bmatrix} cos(m\theta) & -sin(m\theta) \\ sin(m\theta) & cos(m\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

$$= \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}^T \begin{bmatrix} cos(n\theta) & -sin(n\theta) \\ sin(n\theta) & cos(n\theta) \end{bmatrix}^T \begin{bmatrix} cos(m\theta) & -sin(m\theta) \\ sin(m\theta) & cos(m\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

Rotate $-n\theta$ Rotate $m\theta$

$$= \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}^T \begin{bmatrix} cos\big((m-n)\theta\big) & -sin\big((m-n)\theta\big) \\ sin\big((m-n)\theta\big) & cos\big((m-n)\theta\big) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

Rotate$(m-n)\theta$

Rotate $n\theta$

$$\begin{bmatrix} \boldsymbol{k_A^n}[0] \\ \boldsymbol{k_A^n}[1] \end{bmatrix} = \begin{bmatrix} cos(n\theta) & -sin(n\theta) \\ sin(n\theta) & cos(n\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}$$

Rotate $m\theta$

$$\begin{bmatrix} \boldsymbol{q_B^m}[0] \\ \boldsymbol{q_B^m}[1] \end{bmatrix} = \begin{bmatrix} cos(m\theta) & -sin(m\theta) \\ sin(m\theta) & cos(m\theta) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{k_A^n}[0] \\ \boldsymbol{k_A^n}[1] \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{q_B^m}[0] \\ \boldsymbol{q_B^m}[1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}^T \begin{bmatrix} cos\big((m-n)\theta\big) & -sin\big((m-n)\theta\big) \\ sin\big((m-n)\theta\big) & cos\big((m-n)\theta\big) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

Rotate $(m-n)\theta$

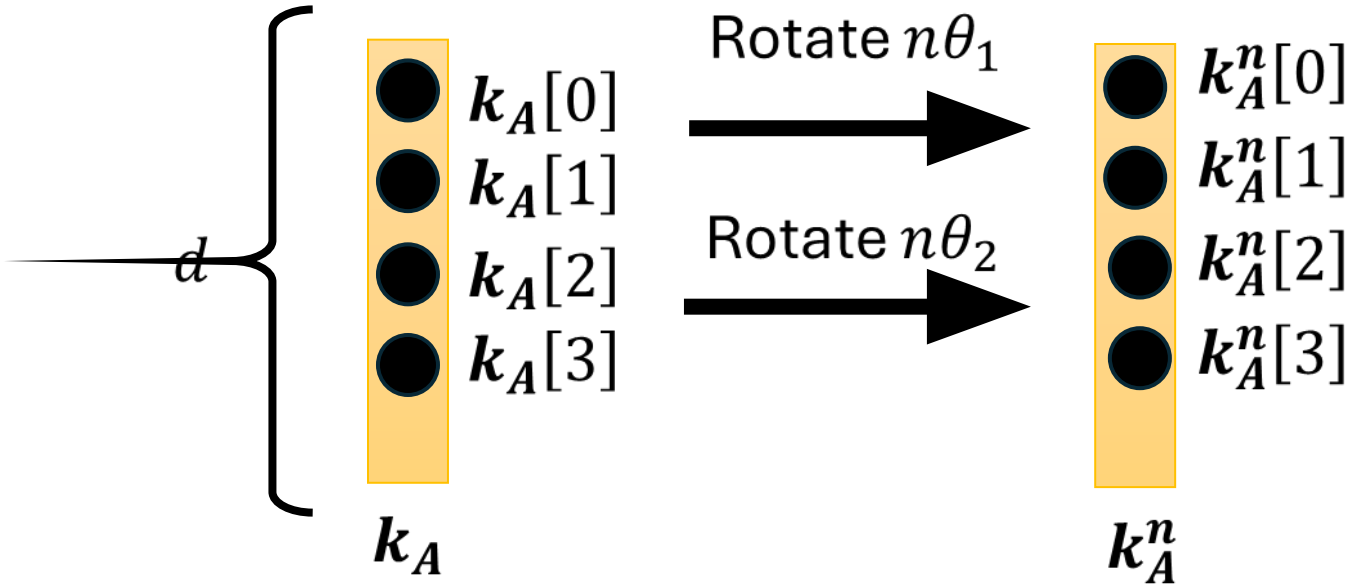
$$\begin{bmatrix} \boldsymbol{k_A^{n+r}}[0] \\ \boldsymbol{k_A^{n+r}}[1] \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{q_B^{m+r}}[0] \\ \boldsymbol{q_B^{m+r}}[1] \end{bmatrix}$$

$$= \begin{bmatrix} \boldsymbol{k_A}[0] \\ \boldsymbol{k_A}[1] \end{bmatrix}^T \begin{bmatrix} cos\Big(\big((m+r)-(n+r)\big)\theta\Big) & -sin\Big(\big((m+r)-(n+r)\big)\theta\Big) \\ sin\Big(\big((m+r)-(n+r)\big)\theta\Big) & cos\Big(\big((m+r)-(n+r)\big)\theta\Big) \end{bmatrix} \begin{bmatrix} \boldsymbol{q_B}[0] \\ \boldsymbol{q_B}[1] \end{bmatrix}$$

Rotate $\big((m+r)-(n+r)\big)\theta$ = Rotate $(m-n)\theta$

旋轉角度怎麼設定

https://arxiv.org/abs/2104.09864



$$\theta_i = \frac{1}{10000^{2i/d}}$$

$$i = 0,1,\ldots,\frac{d}{2} - 1$$

RoPE 沒有越遠越小

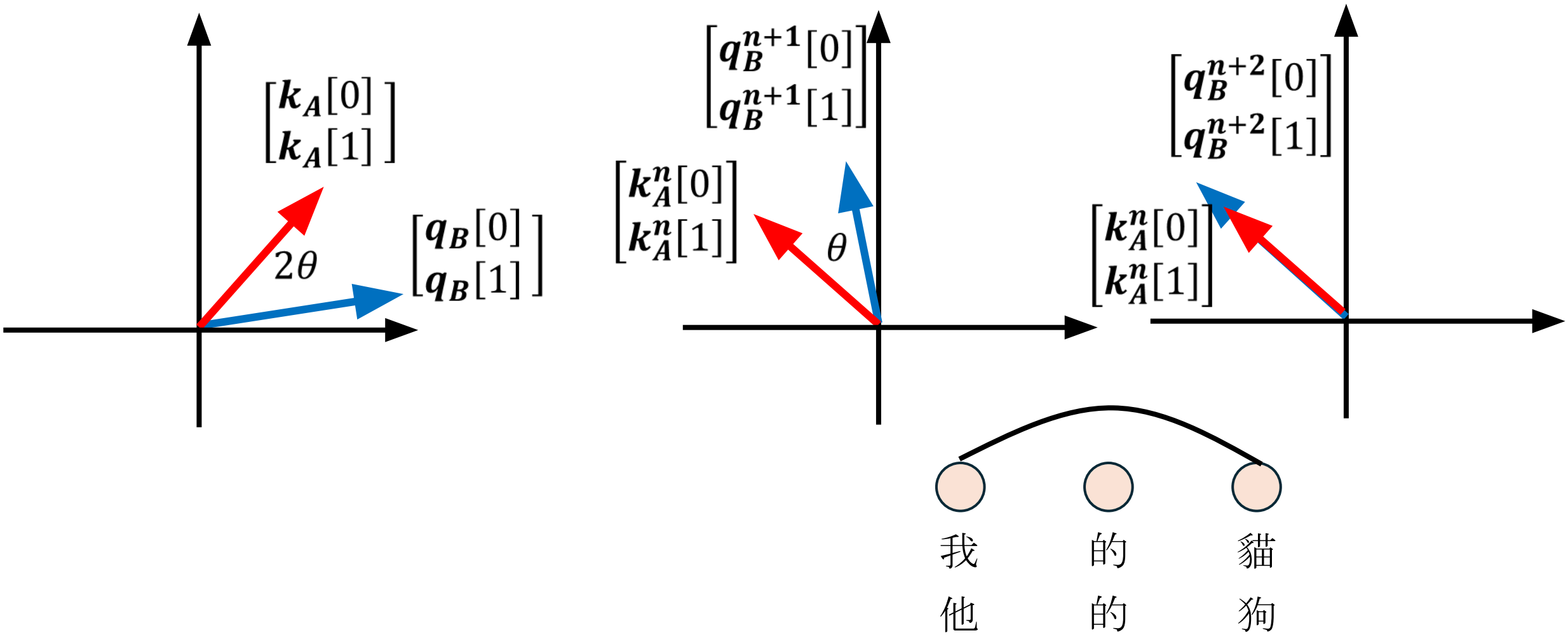
https://arxiv.org/pdf/2410.06205

$$\begin{bmatrix} \boldsymbol{k}_A^n[0] \\ \boldsymbol{k}_A^n[1] \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{q}_B^m[0] \\ \boldsymbol{q}_B^m[1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{k}_A[0] \\ \boldsymbol{k}_A[1] \end{bmatrix}^T \begin{bmatrix} cos\big((m-n)\theta\big) & -sin\big((m-n)\theta\big) \\ sin\big((m-n)\theta\big) & cos\big((m-n)\theta\big) \end{bmatrix} \begin{bmatrix} \boldsymbol{q}_B[0] \\ \boldsymbol{q}_B[1] \end{bmatrix}$$

Rotate $(m-n)\theta$

https://colab.research.google.com/drive/1rWDtAkScrb2K3tcprSTzwuRQo5bGiyKJ?usp=sharing

RoPE 沒有越遠越小，不一定是一件壞事

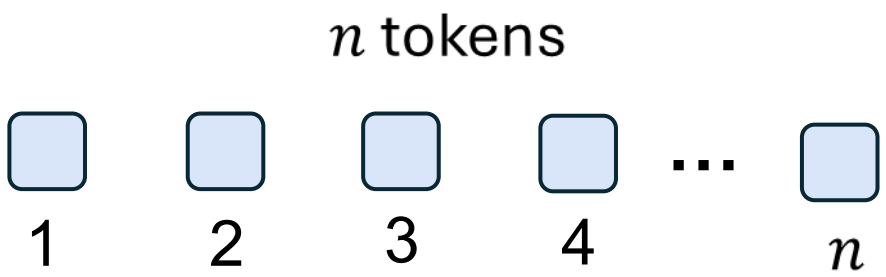


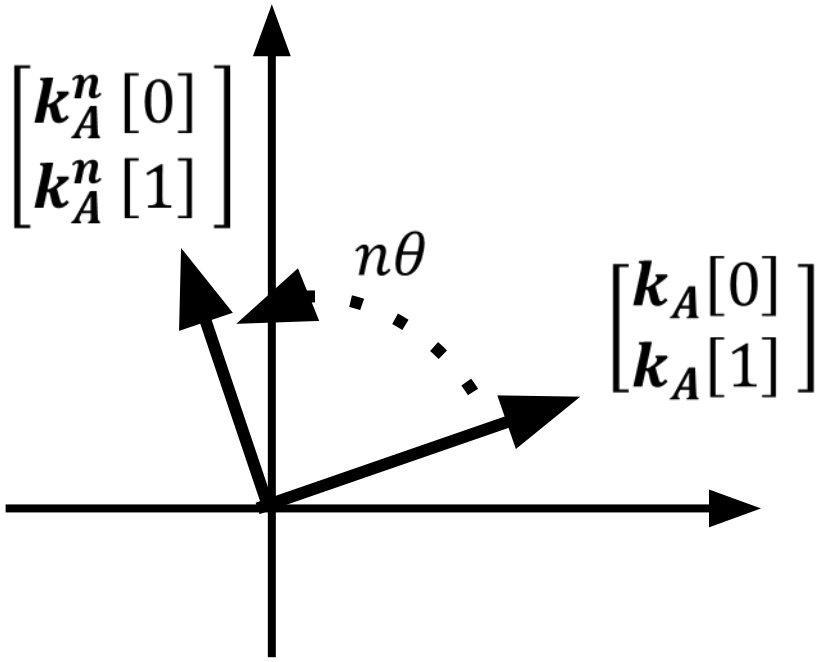
Train Short, Test Long

Reference:
https://amaarora.github.io/posts/2025-09-21-rope-context-extension.html

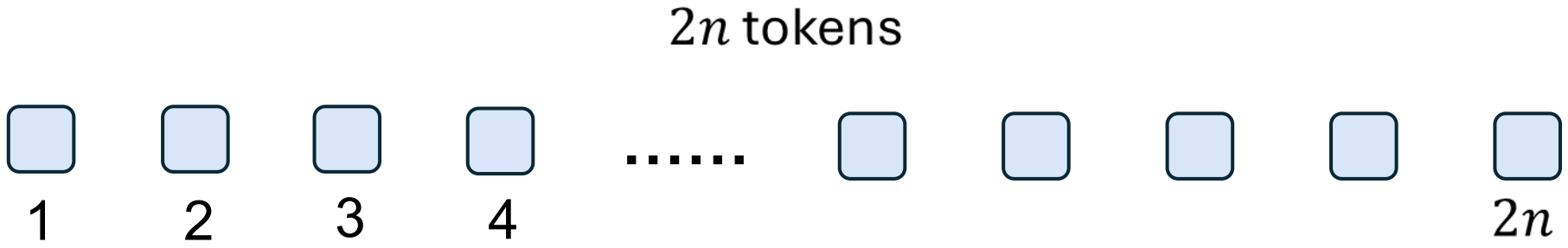
Train Short, Test Long

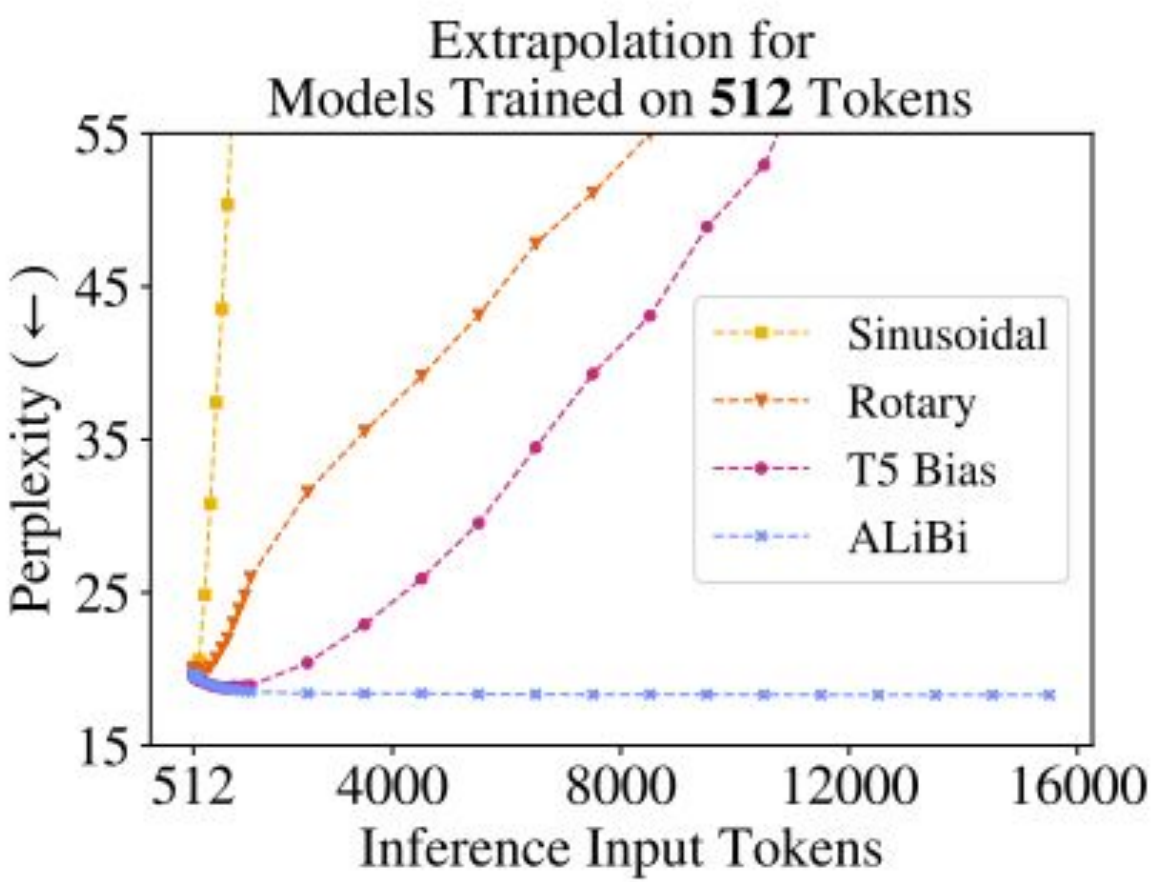
Train

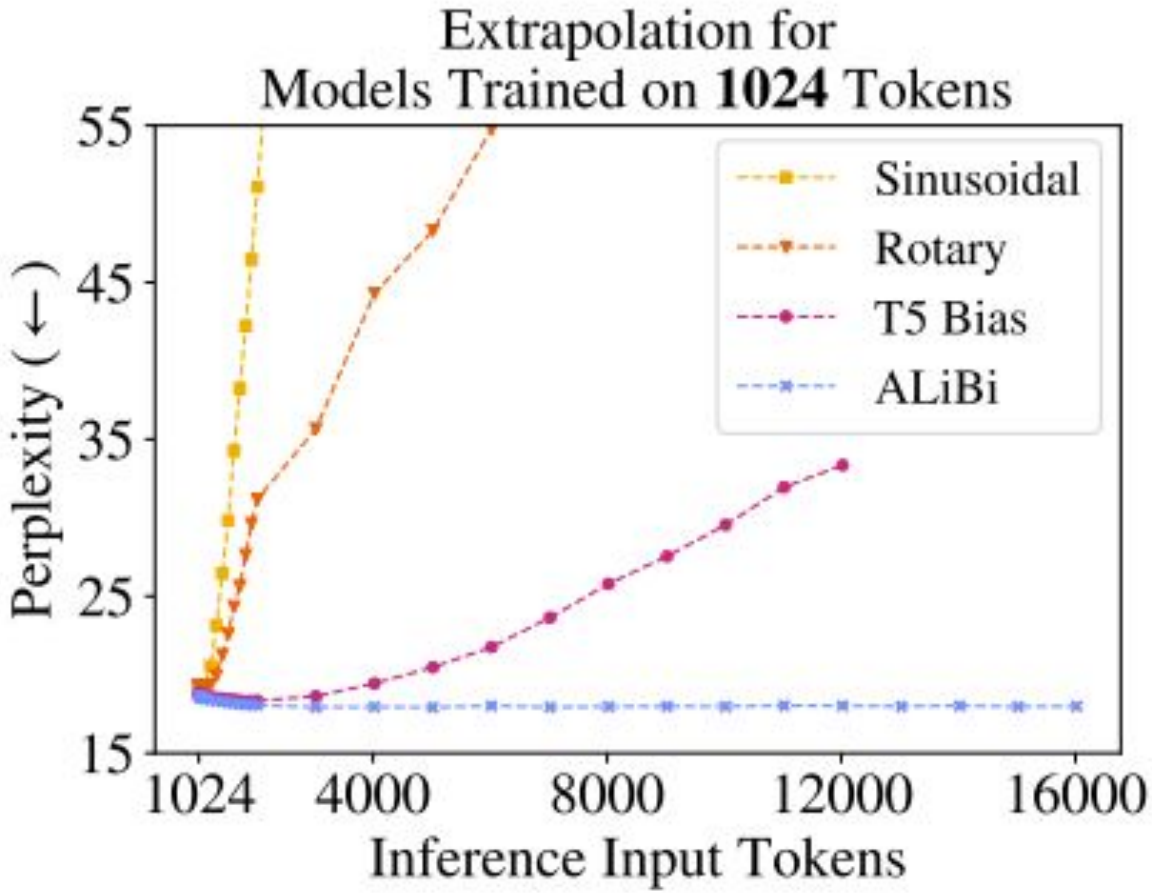




Test

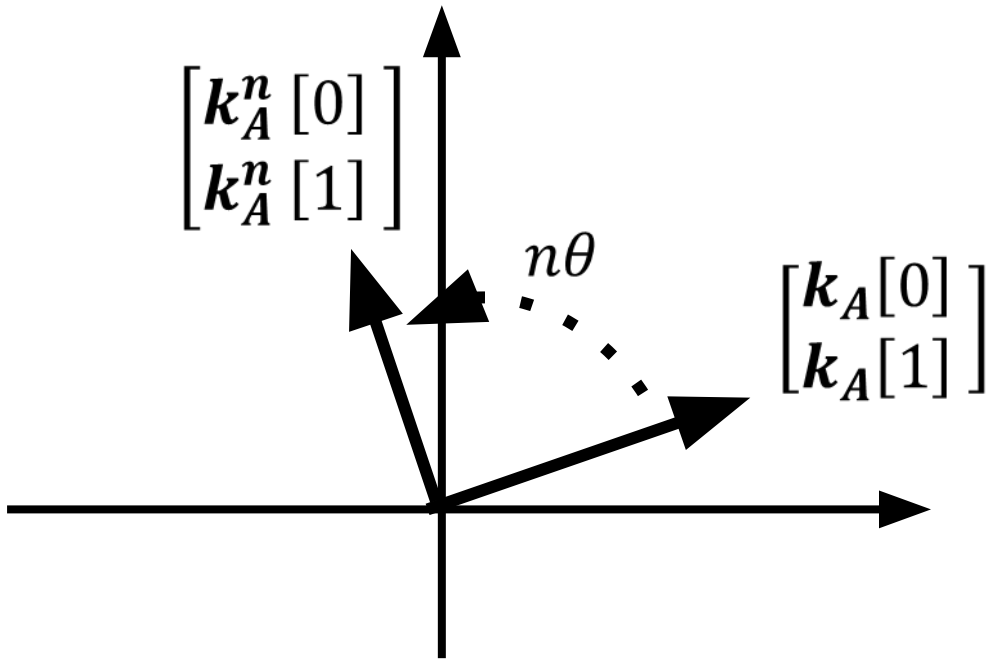




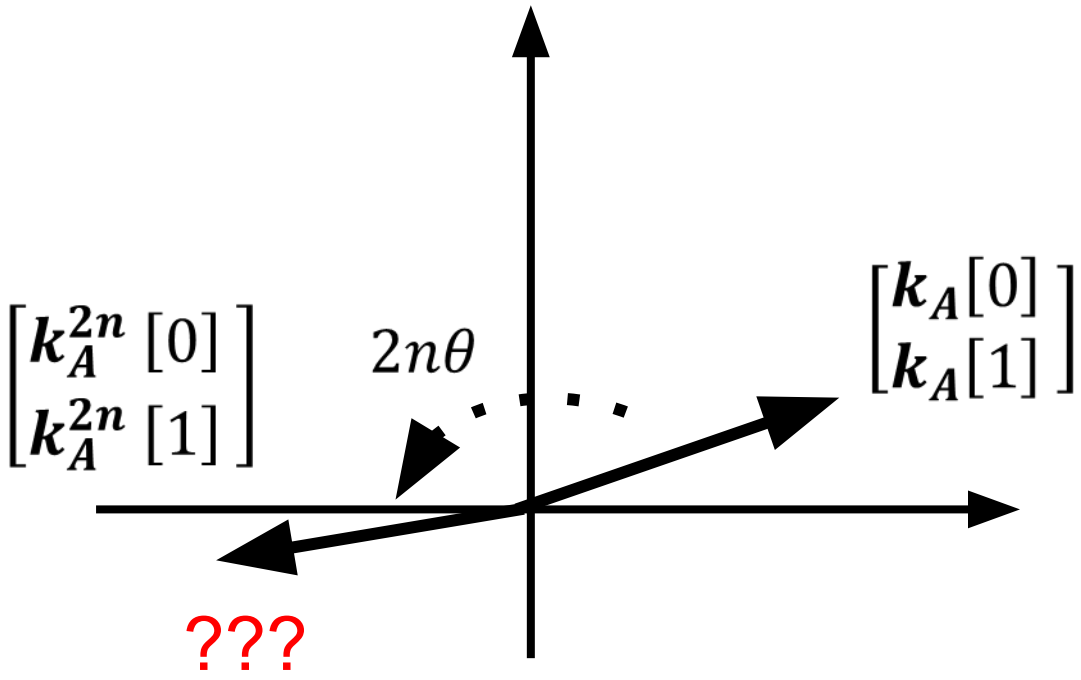


https://arxiv.org/abs/2108.12409

Why Fail?



Training: rotate at most $n\theta$



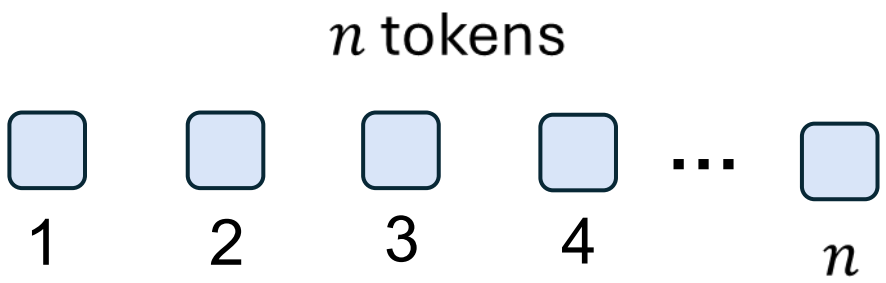
Testing: rotate $2n\theta$

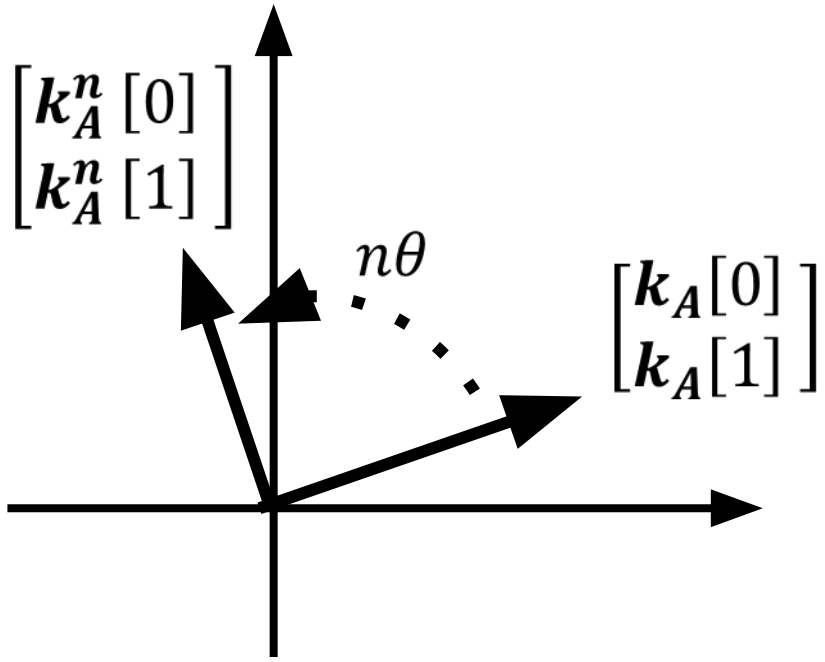
https://arxiv.org/pdf/2306.15595
https://kaiokendev.github.io/context#a-bigger-problem

Position Interpolation

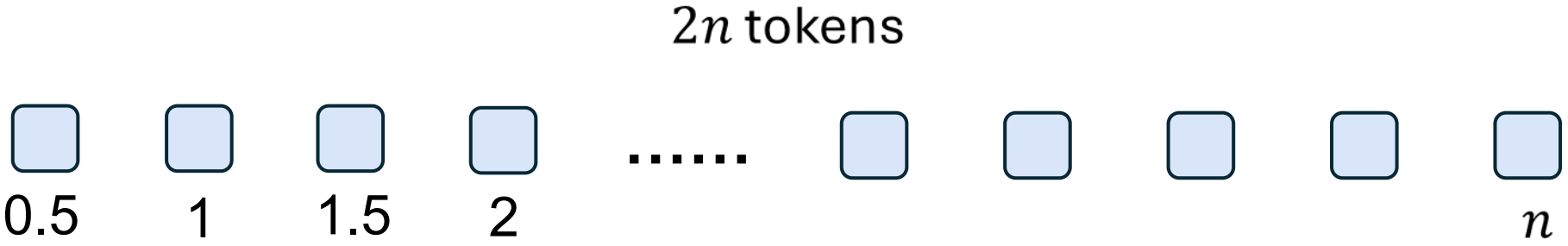
仍然需要微調模型參數

Train





Test

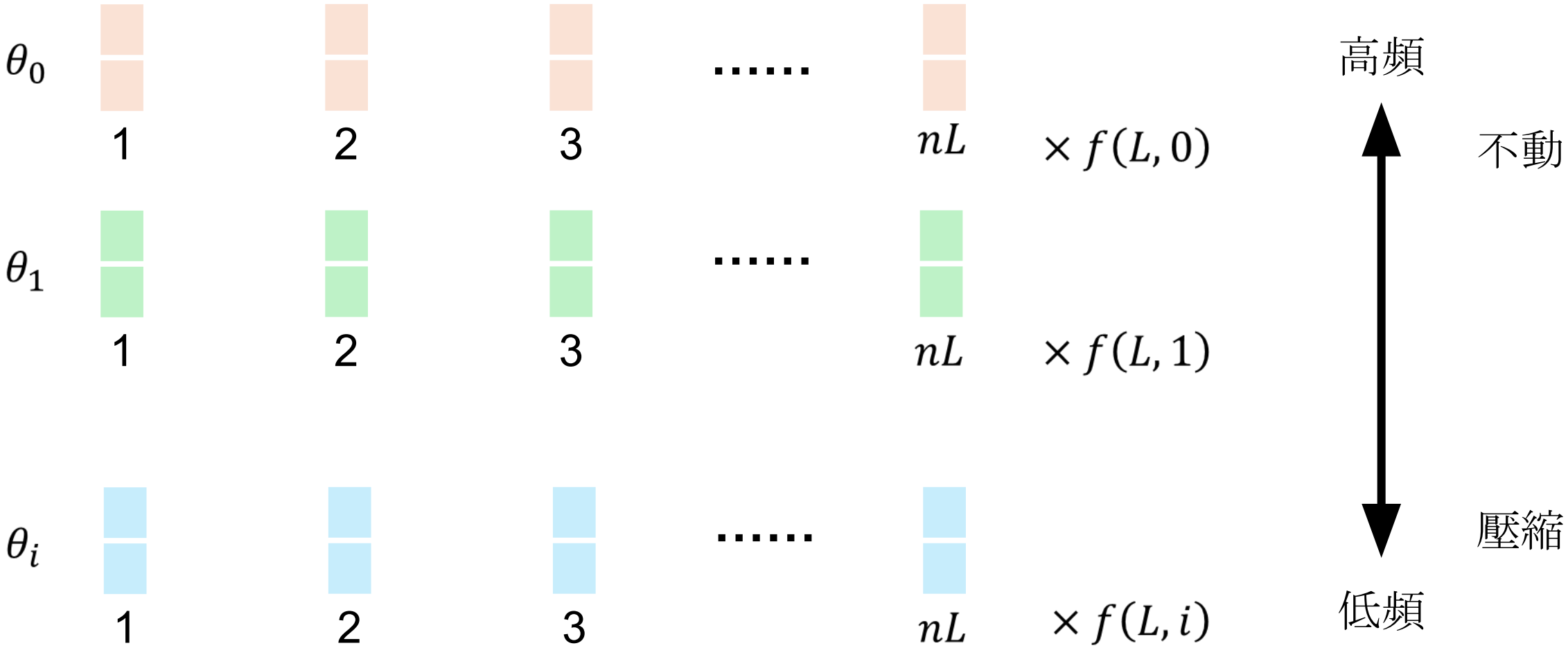


伸長 L 倍，position index 乘上 $1/L$

Frequency-Based Approach

$$\theta_i = \frac{1}{10000^{2i/d}}$$

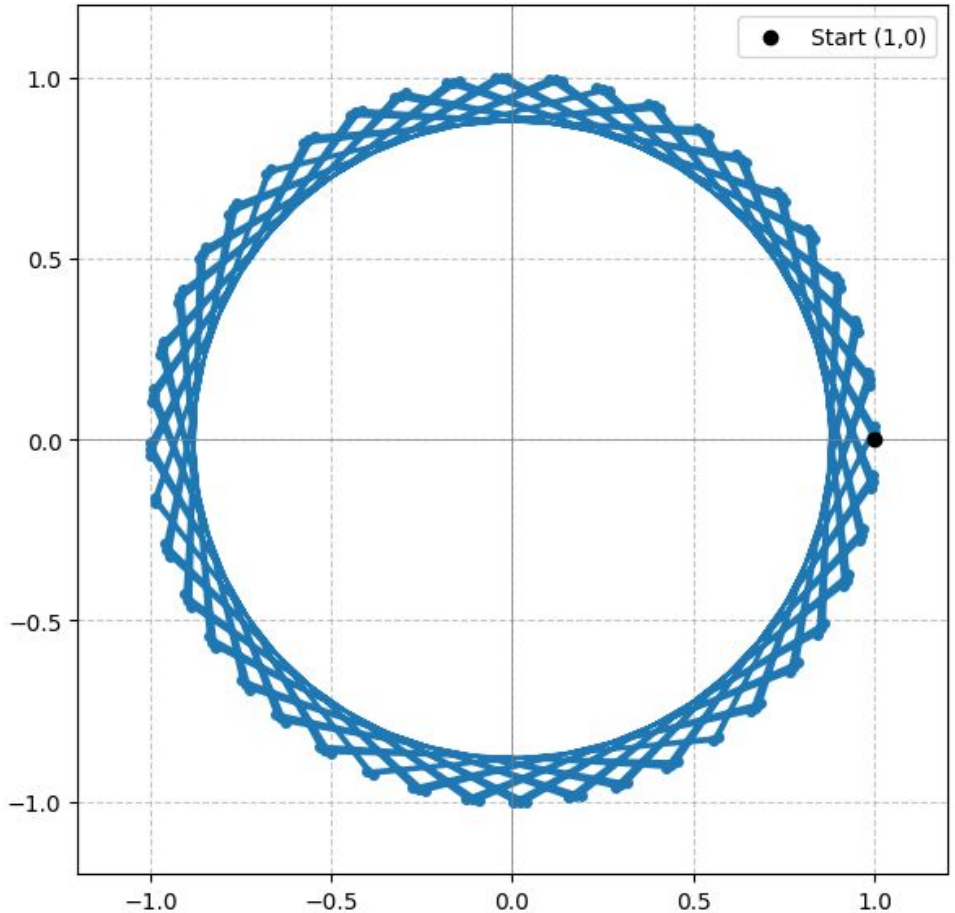
$$i = 0,1,\ldots,\frac{d}{2} - 1$$



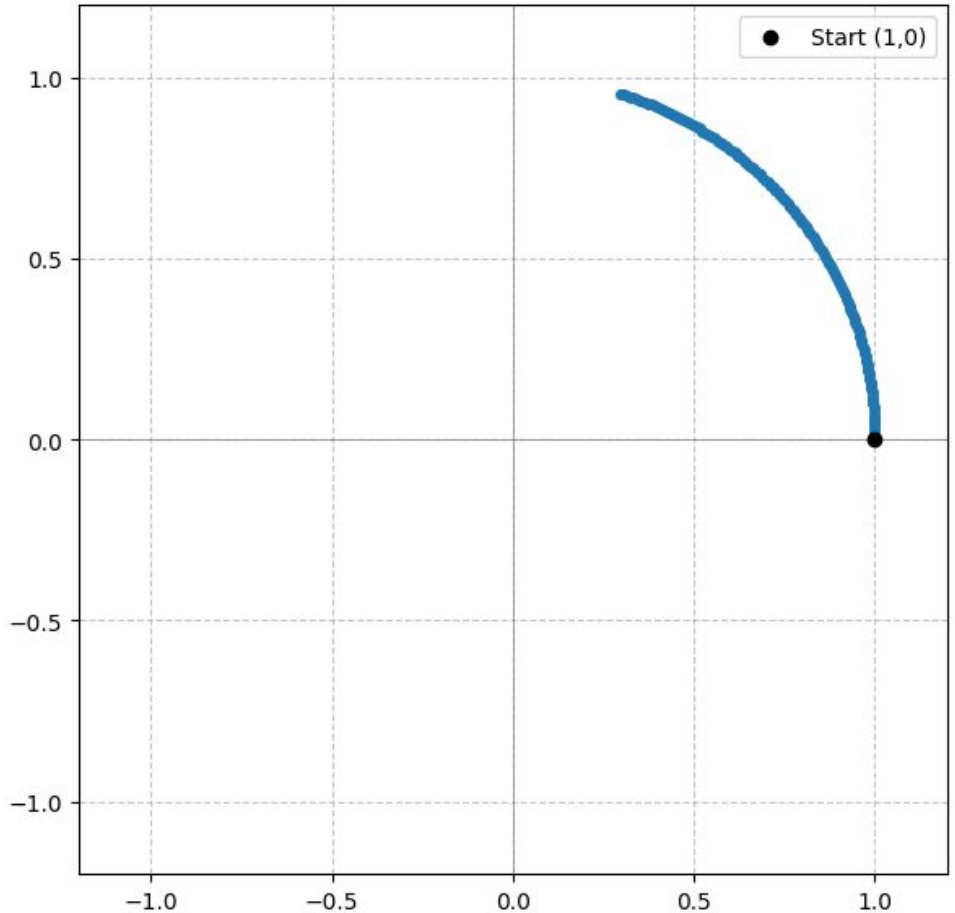
Frequency-Based Approach

$$\theta_i = \frac{1}{10000^{2i/d}}$$

$$i = 0,1,\ldots,\frac{d}{2} - 1$$



θ_0



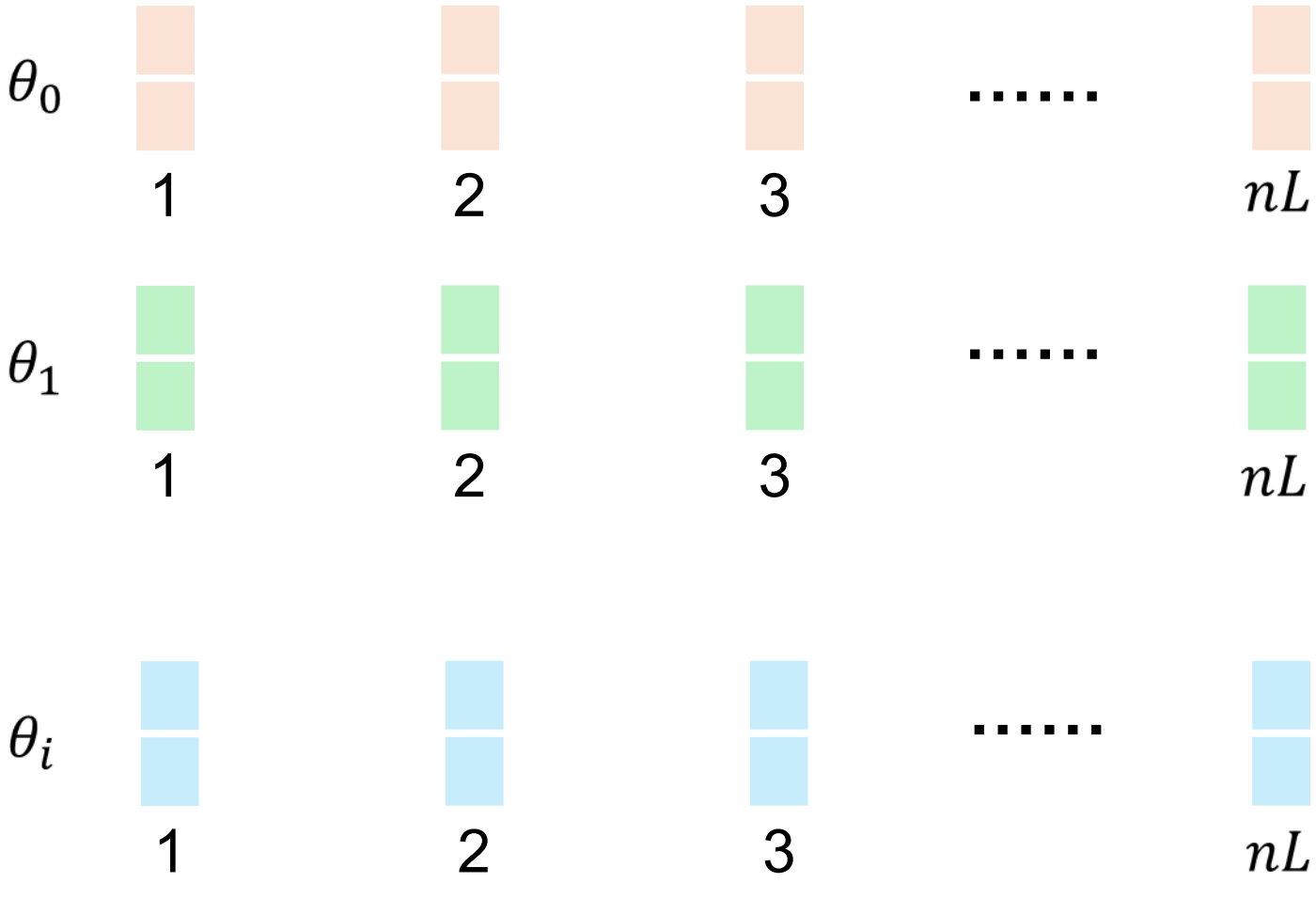
θ_{32}

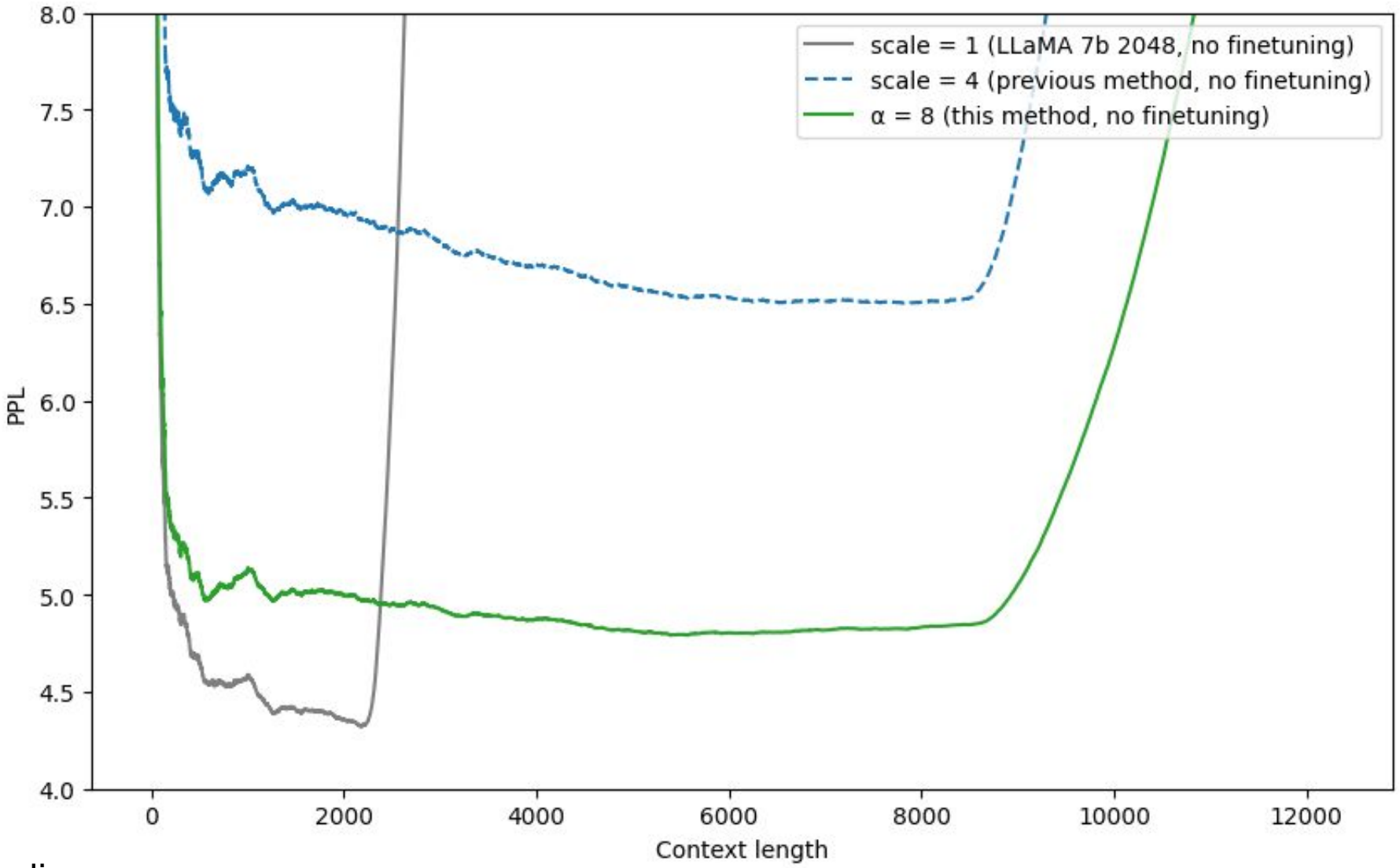
Frequency-Based Approach

NTK-Aware Scaling

$$\theta_i = \frac{1}{10000^{2i/d}}$$

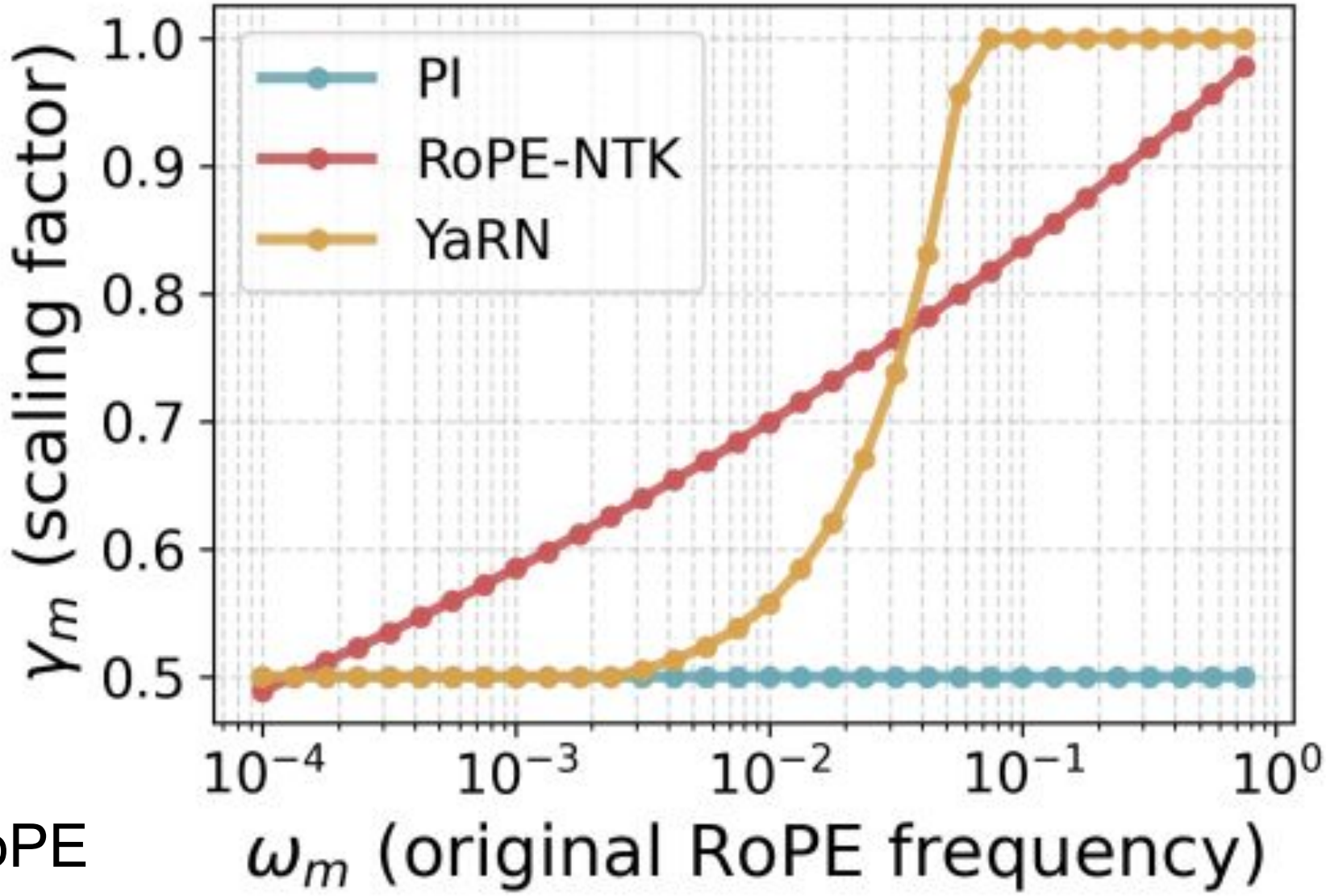
$$i = 0,1,\dots,\frac{d}{2}-1$$





NTK-Aware Scaling

https://www.reddit.com/r/LocalLLaMA/comments/14lz7j5/ntkaware_scaled_rope_allows_llama_models_to_have/



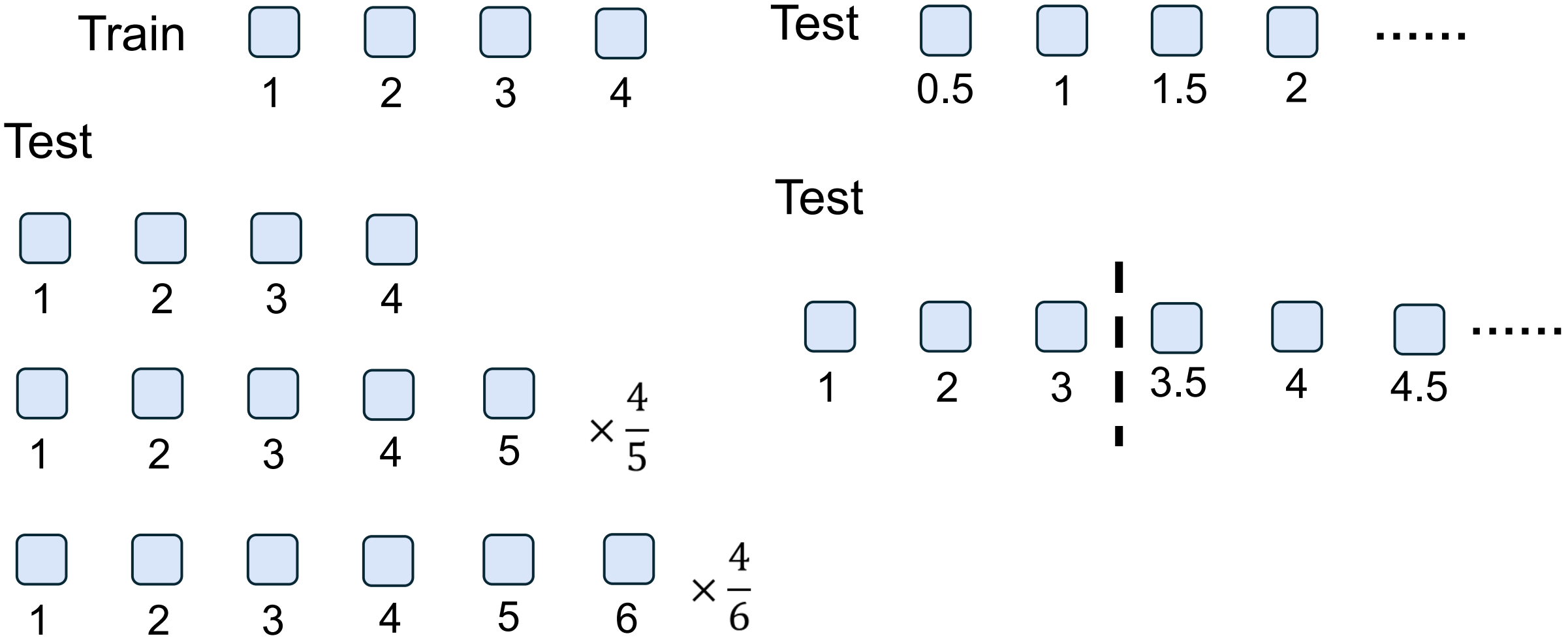
YaRN (Yet another RoPE extensioN method)

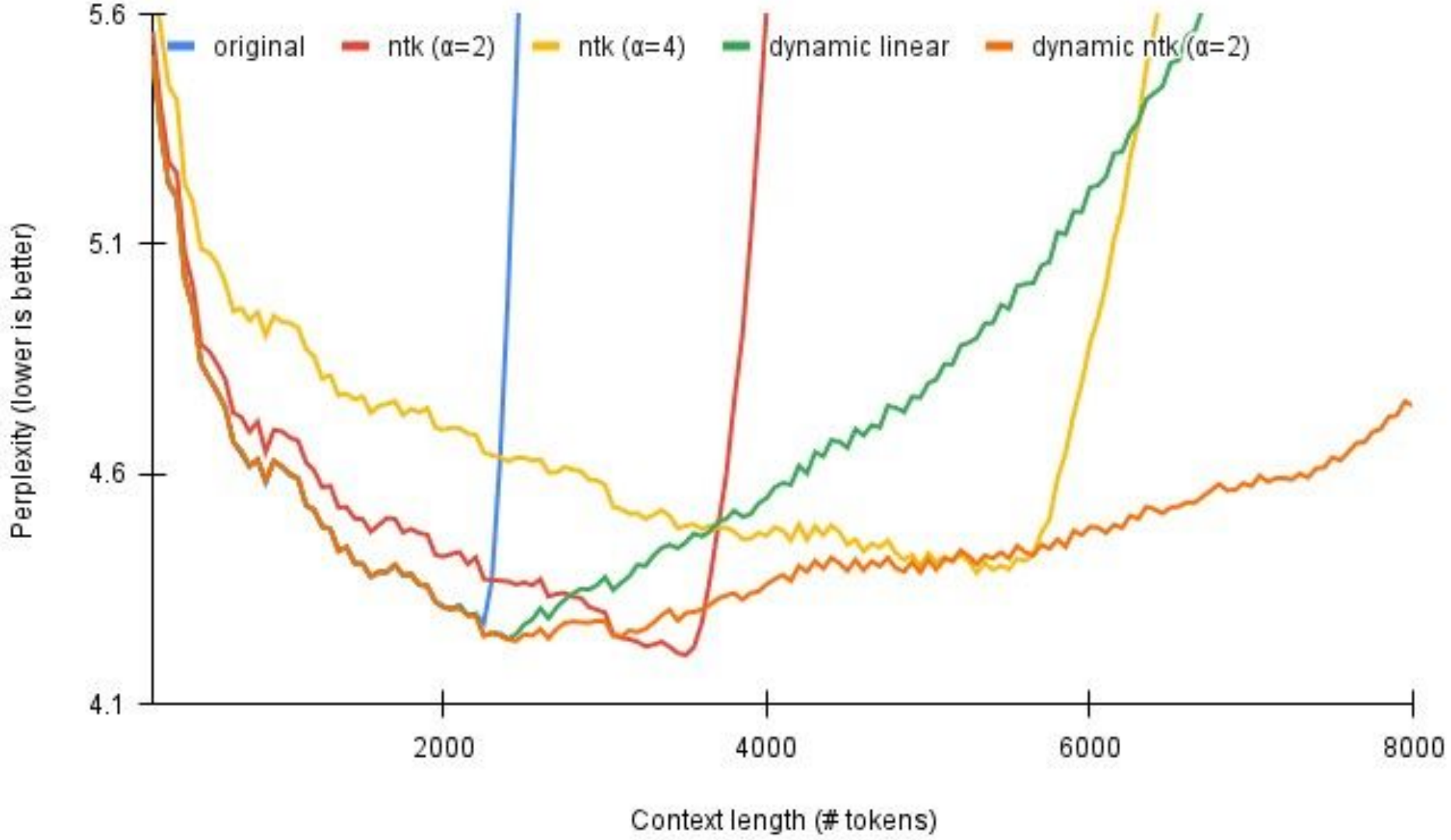
https://arxiv.org/abs/2309.00071

Source of figure: https://arxiv.org/pdf/2512.12167

Dynmaic Scaling

https://www.reddit.com/r/LocalLLaMA/comments/14mrgpr/dynamically_scaled_rope_further_increases/





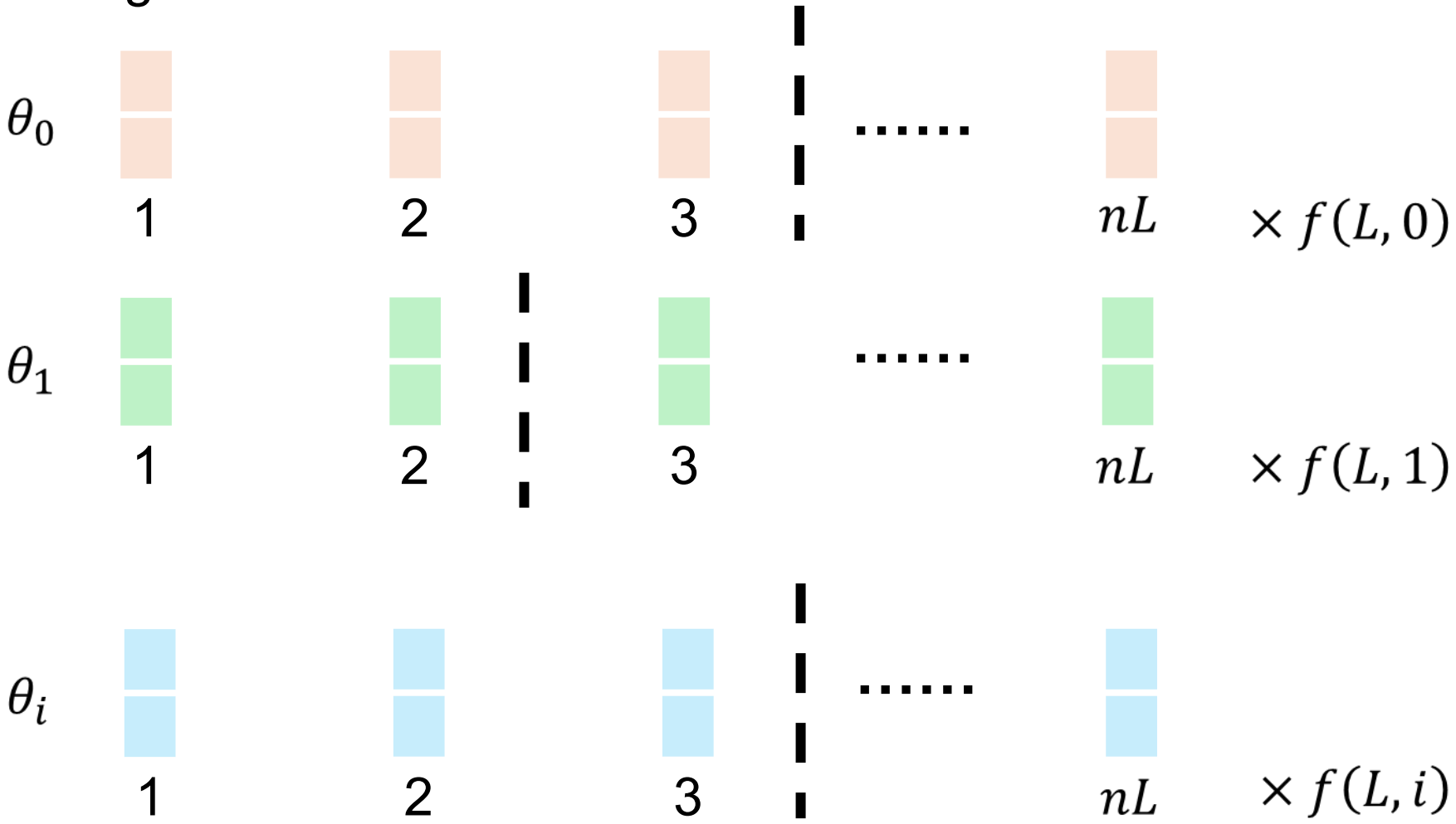
https://www.reddit.com/r/LocalLLaMA/comments/14mrgpr/dynamically_scaled_rope_further_increases/

Frequency-Based + Dynmaic

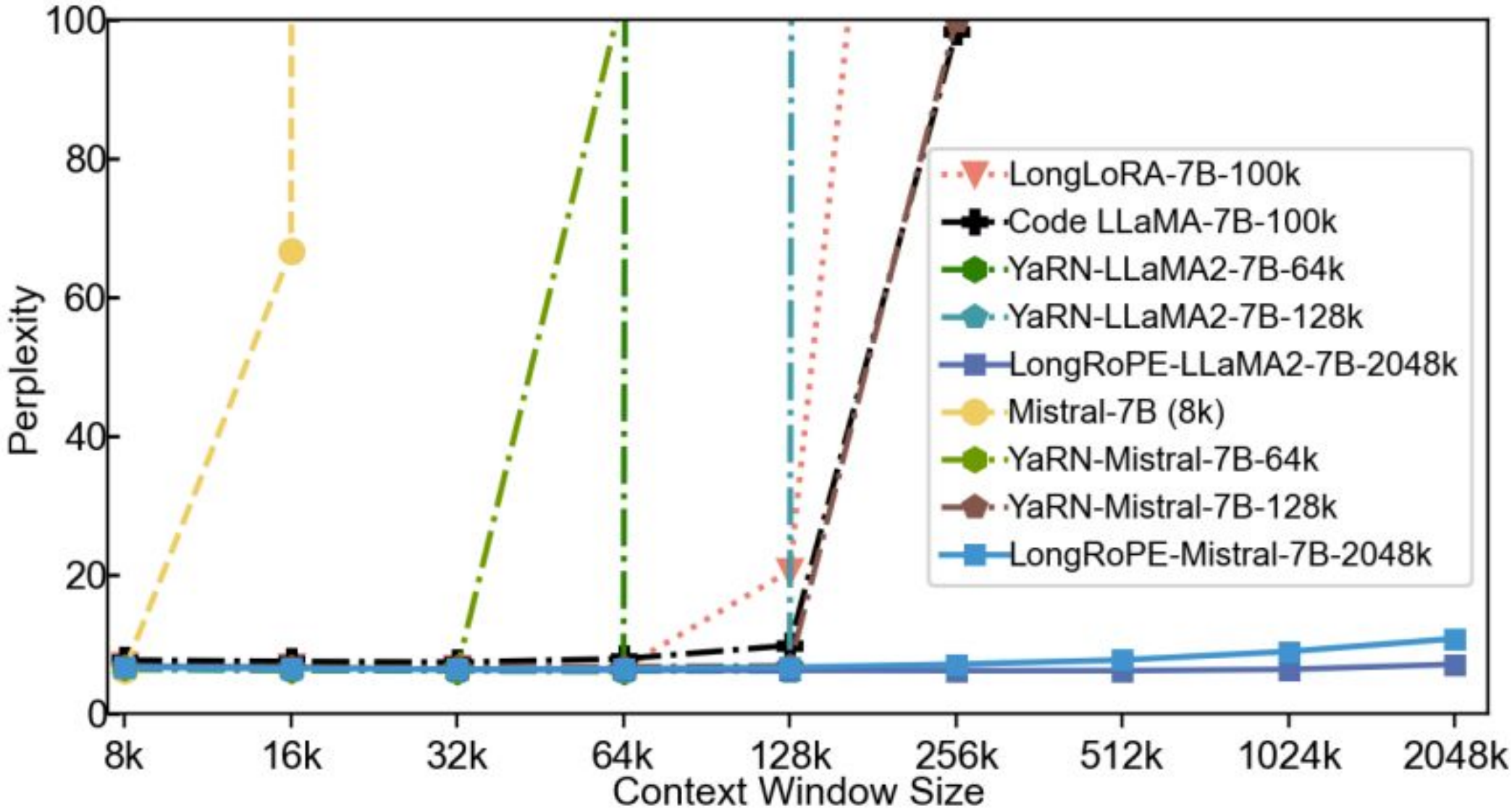
$$\theta_i = \frac{1}{10000^{2i/d}}$$

$$i = 0,1,\ldots,\frac{d}{2} - 1$$

LongRoPE



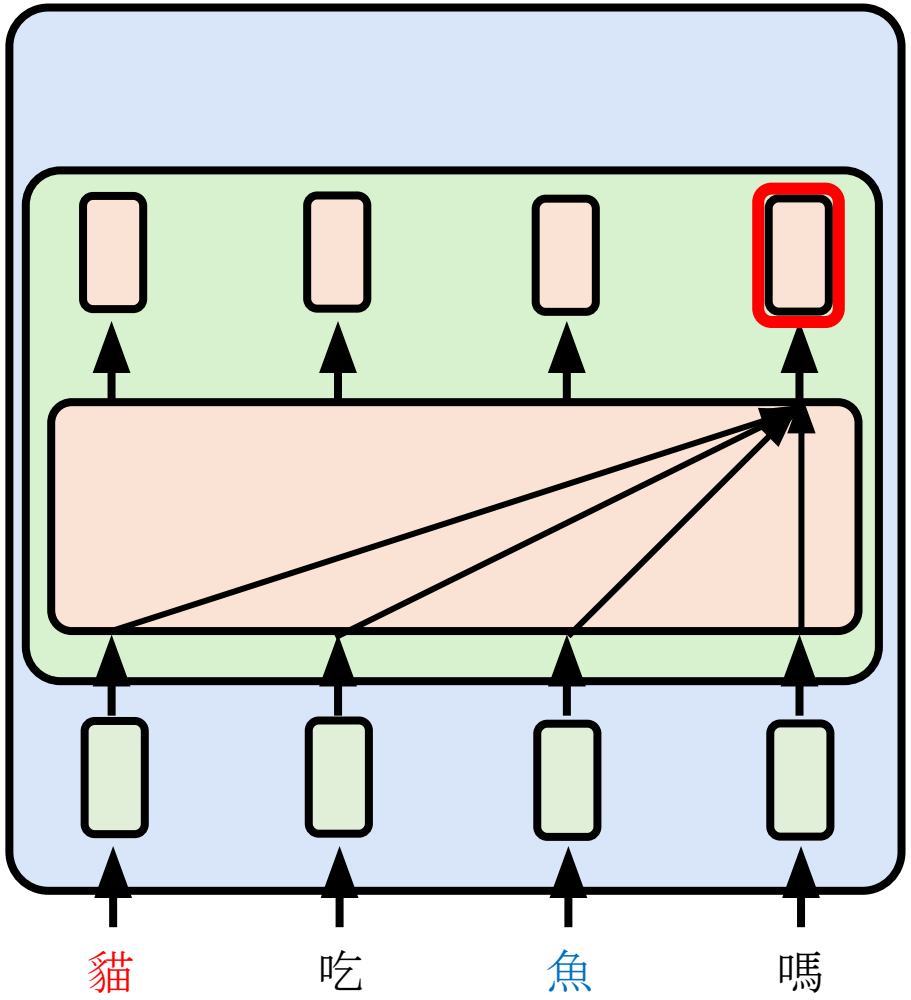
Use evolutionary search

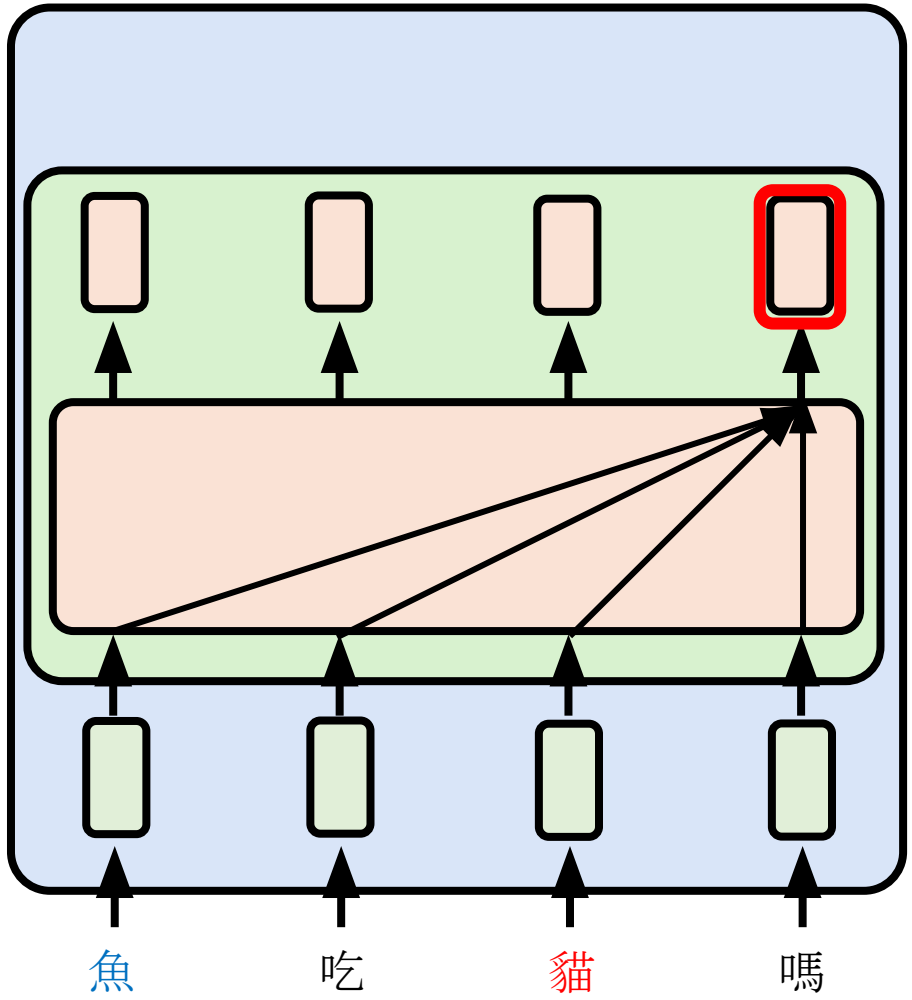


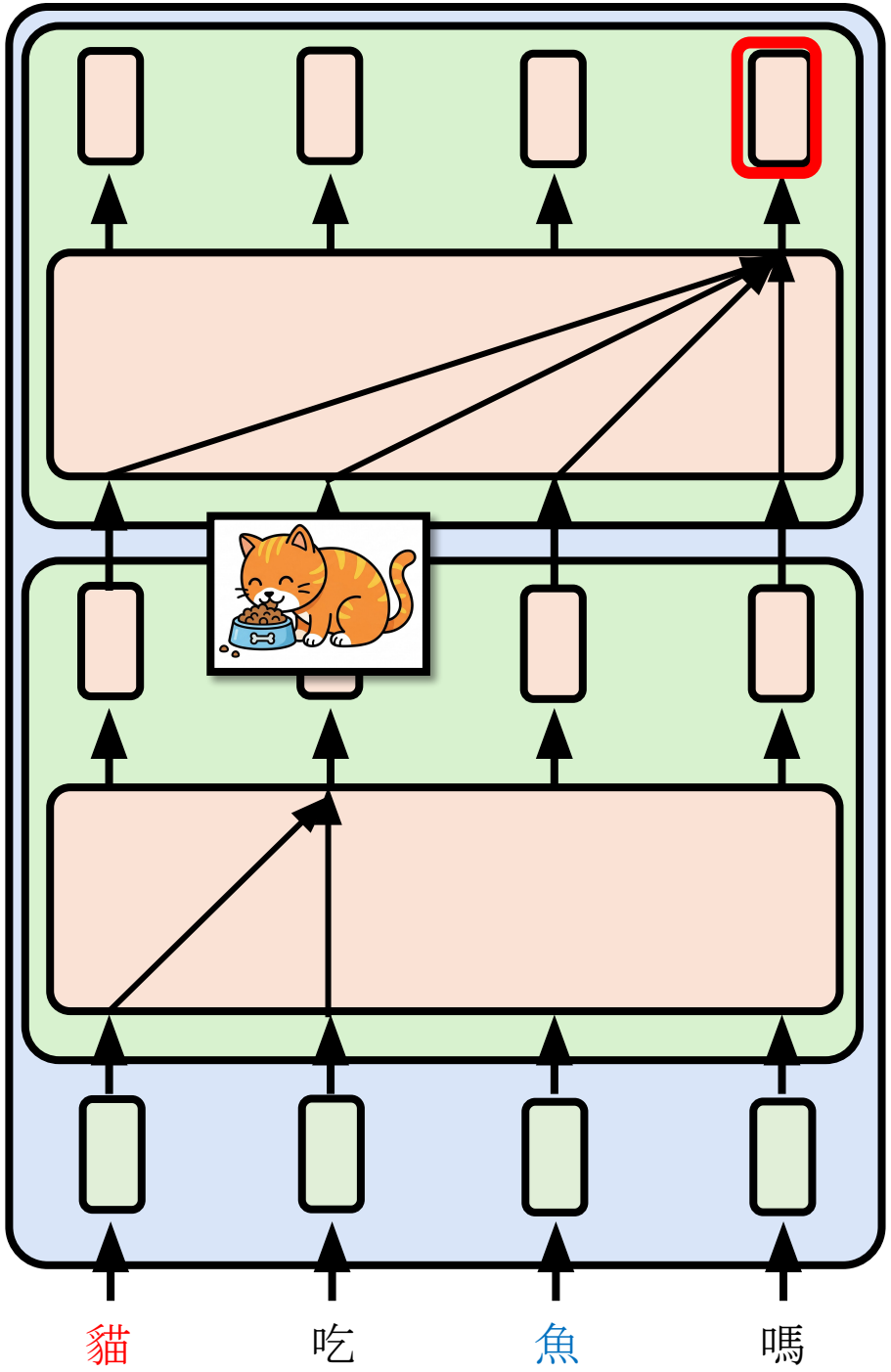
https://arxiv.org/abs/2402.13753

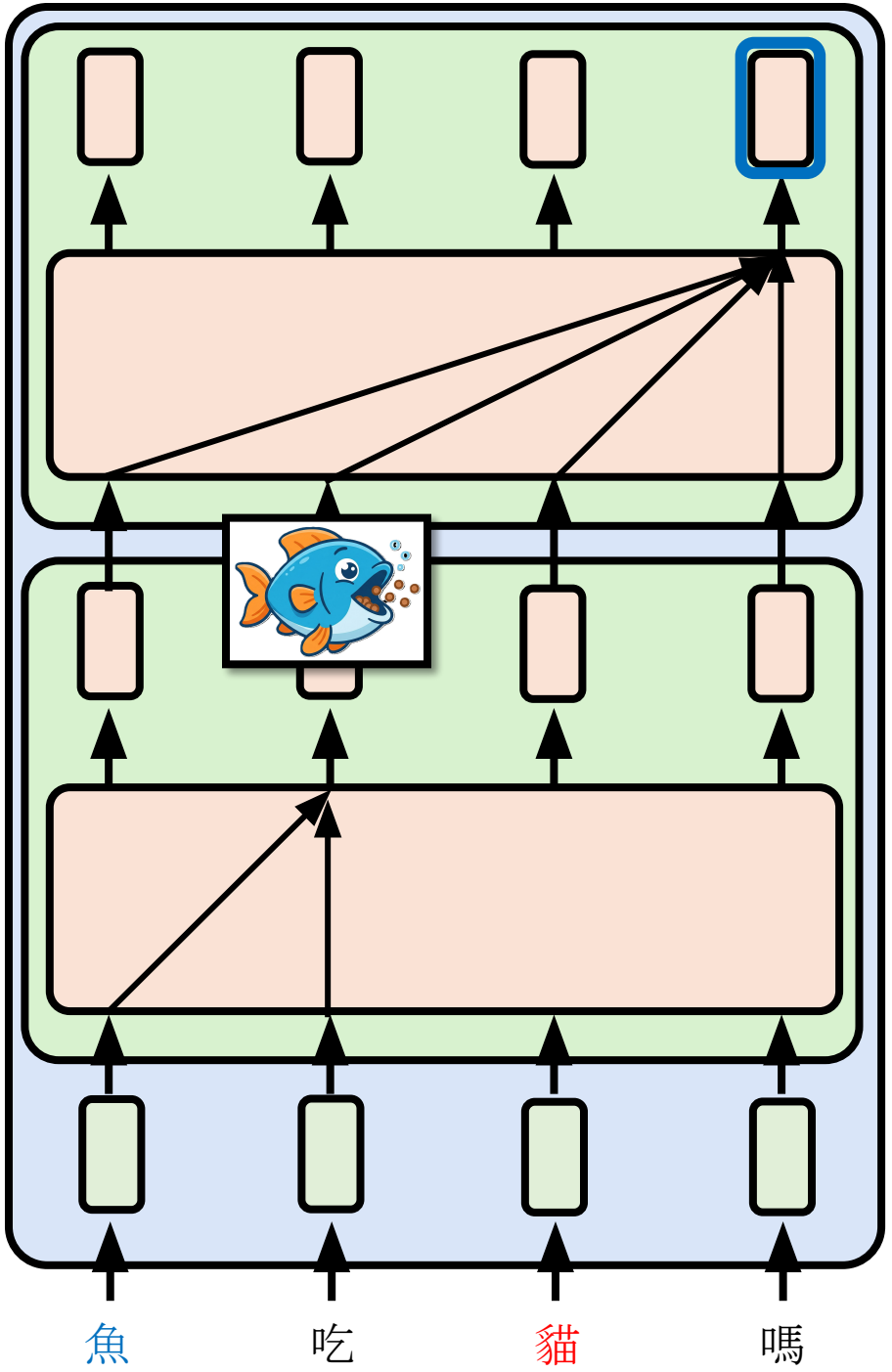
No Positional Embedding?!

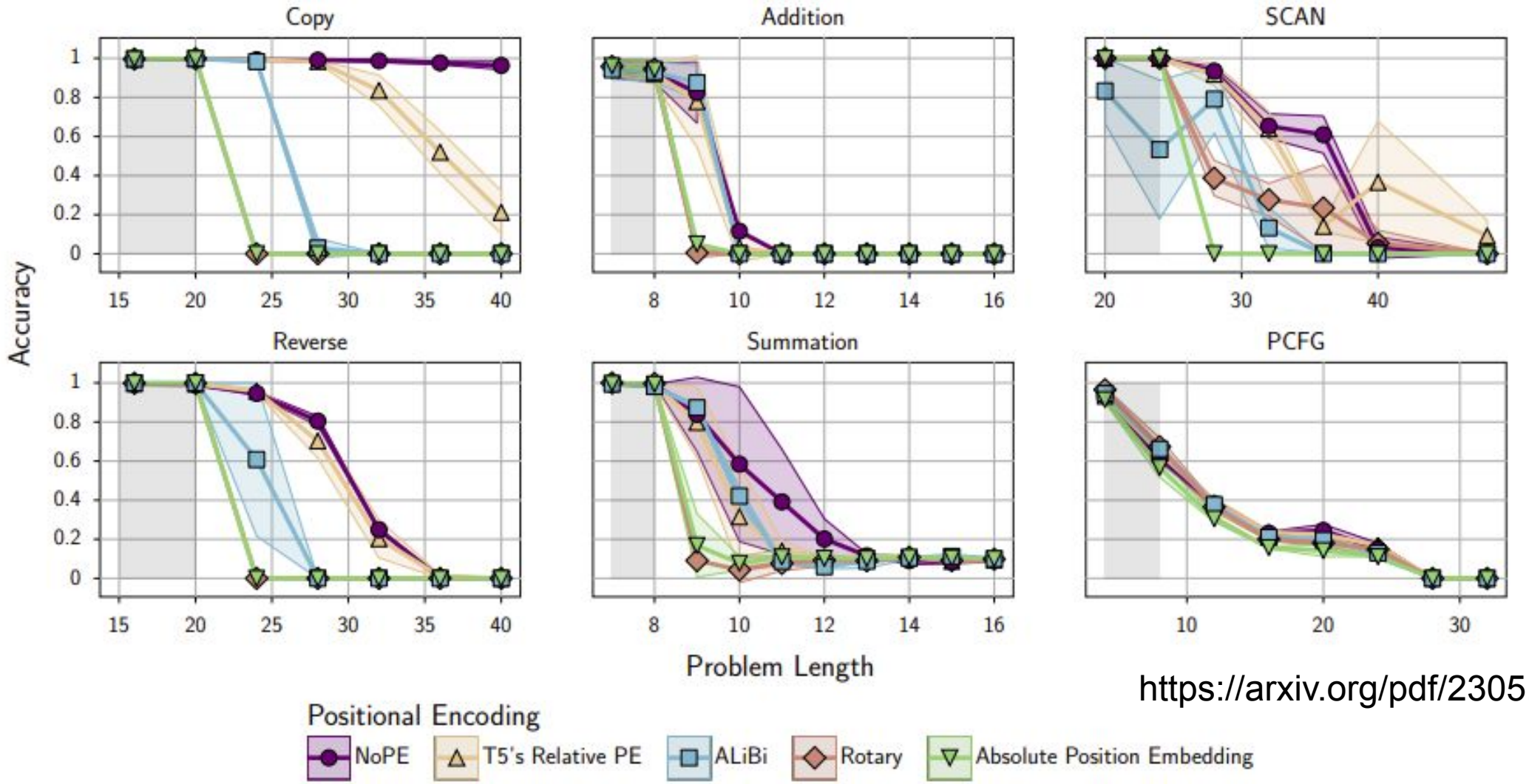
Self-attention 真的沒有位置資訊？



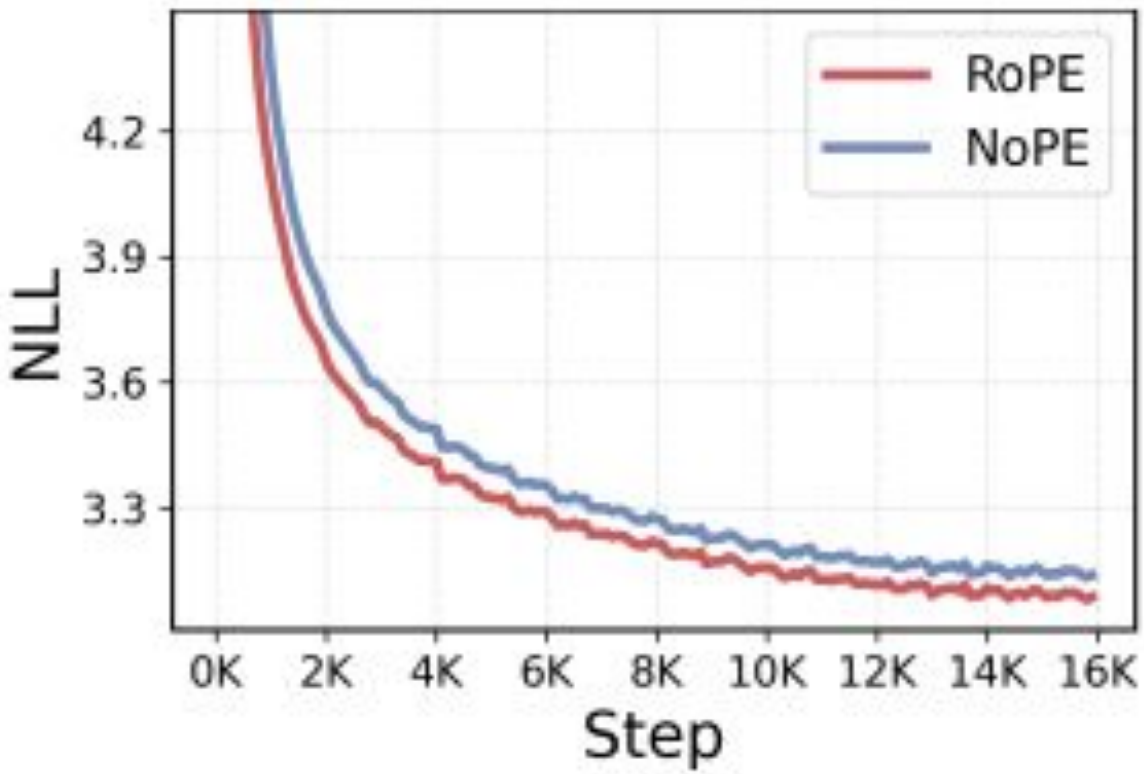


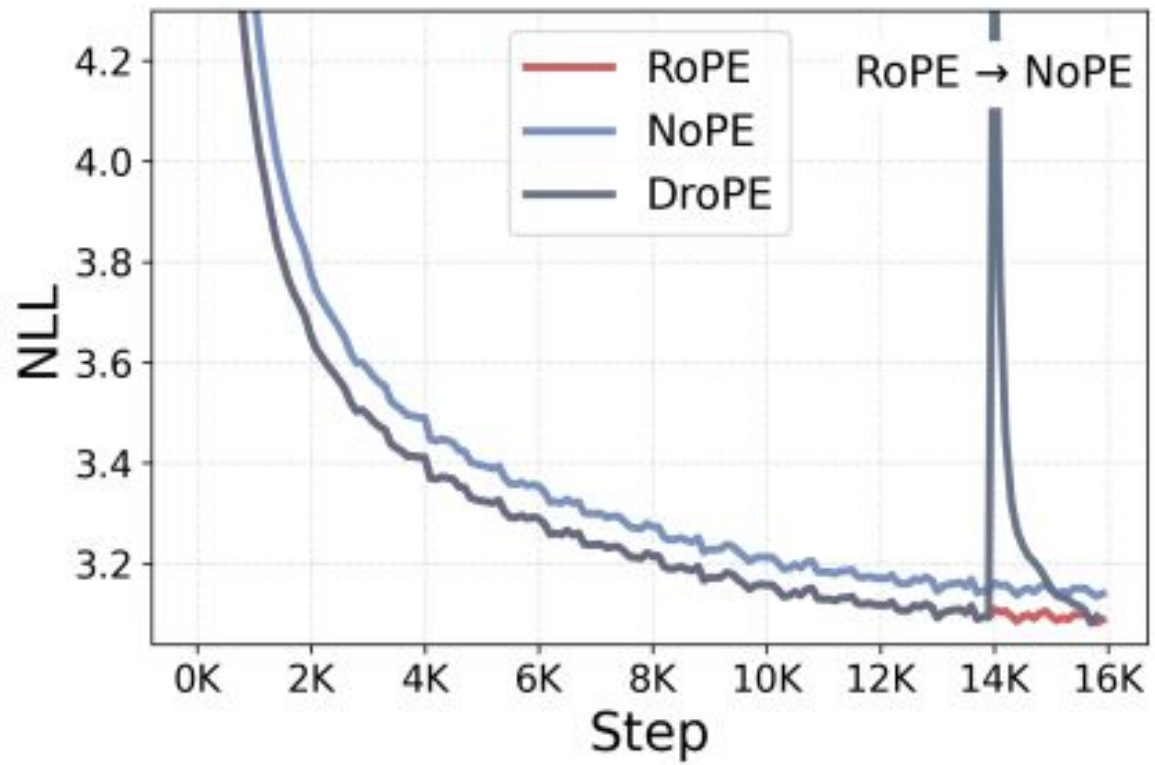

貓
吃
魚
嗎


魚
吃
貓
嗎

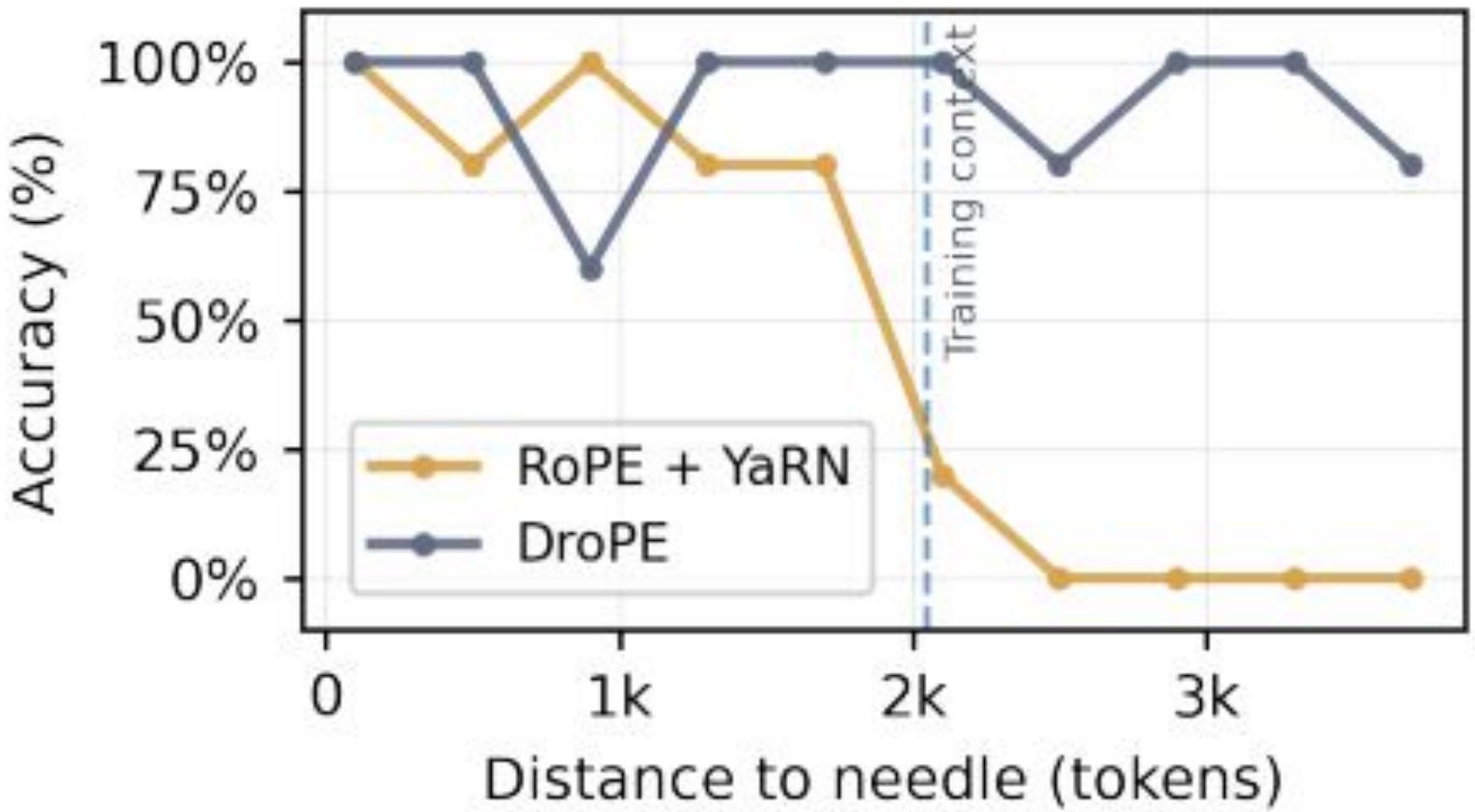


https://arxiv.org/pdf/2305.19466





https://arxiv.org/pdf/2512.12167



https://arxiv.org/pdf/2512.12167

Concluding Remarks

- Absolute Positional Embedding
- Relative Positional Embedding
- RoPE
- Train short, test long
- No Positional Embedding?!