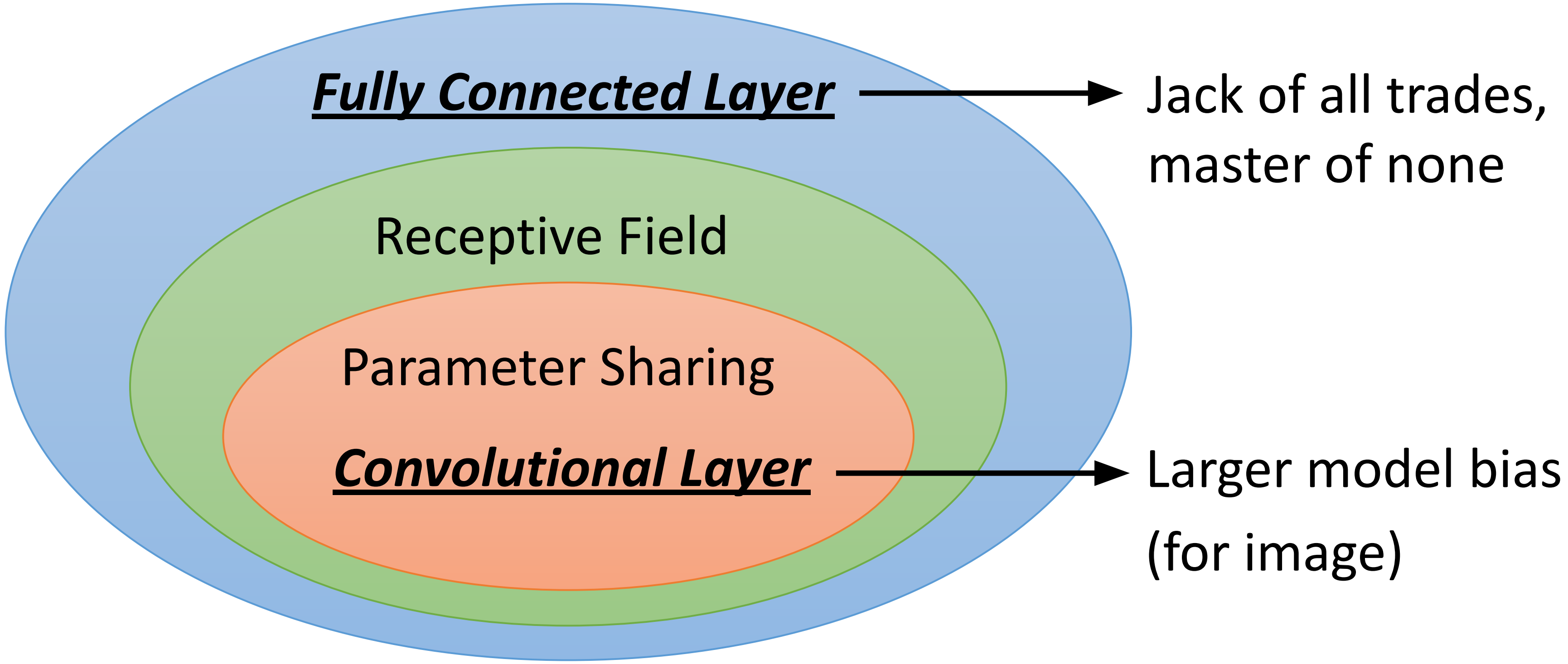
Transformer 的競爭者們

每一種架構的存在都有一個理由

- CNN 存在的理由是什麼？

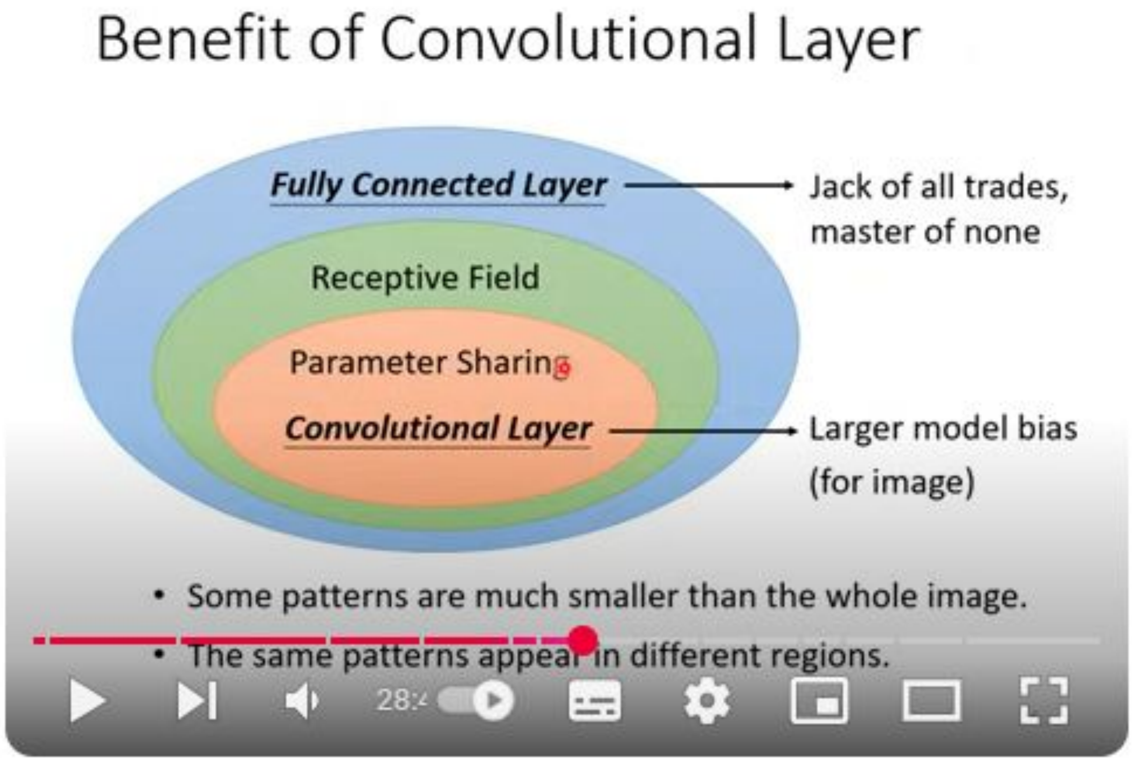


根據影像的特性, 減少不必要的參數, **避免 Overfitting**

每一種架構的存在都有一個理由

- CNN 存在的理由是什麼？

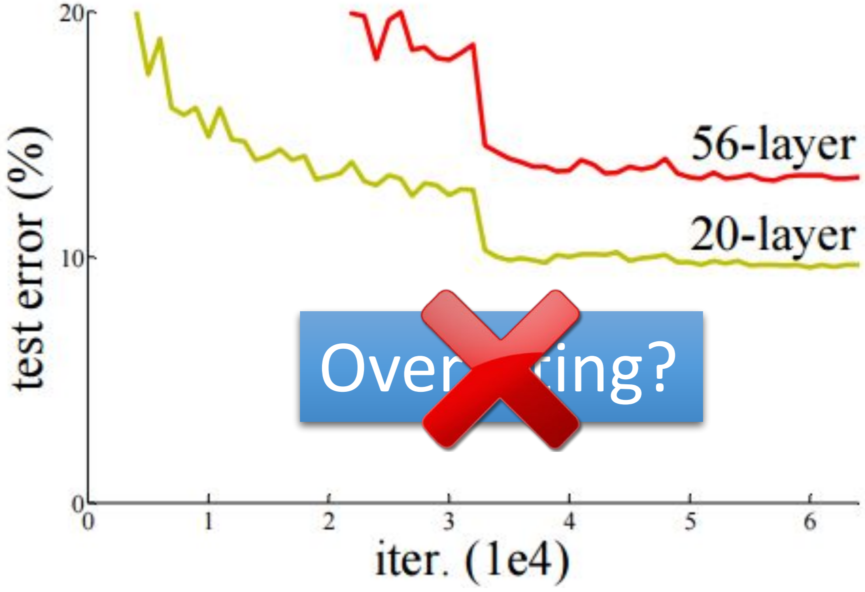
https://youtu.be/OP5HcXJg2Aw?si=RPfmHhsrMtuN0QS6



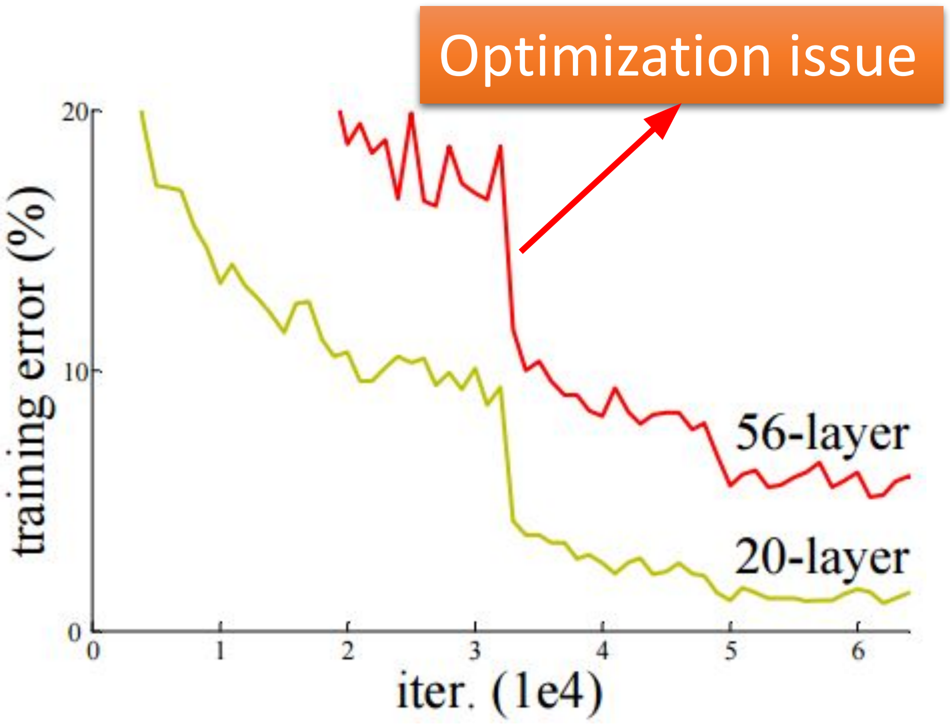
【機器學習2021】卷積神經網路 (Convolutional Neural Networks, CNN)

每一種架構的存在都有一個理由

- Residual Connection 存在的理由是什麼？



Testing Data

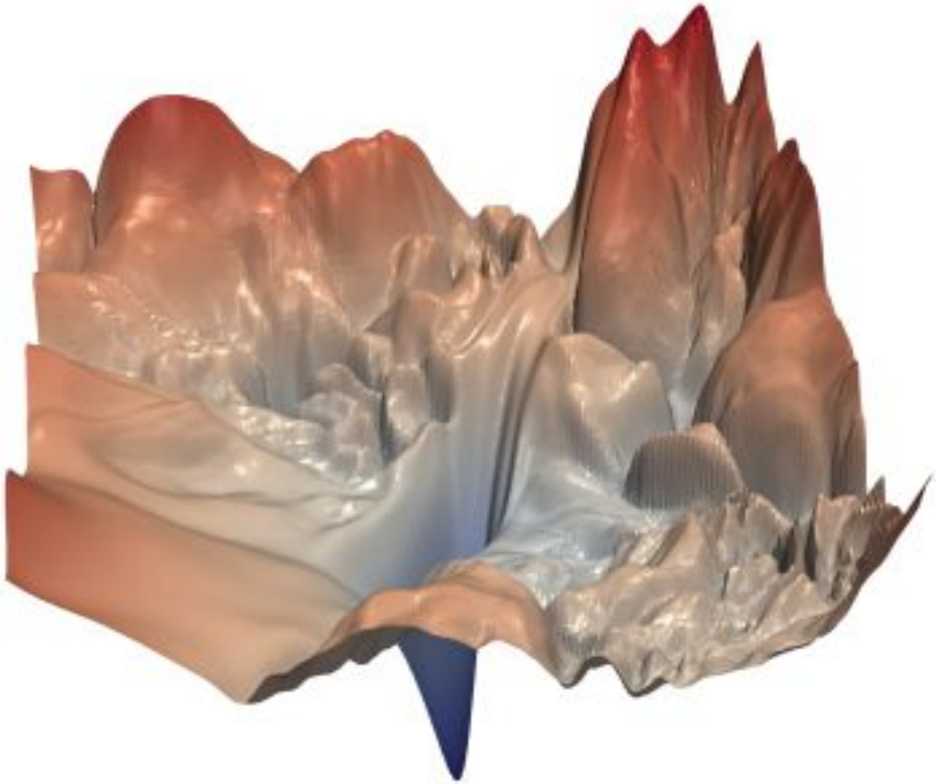


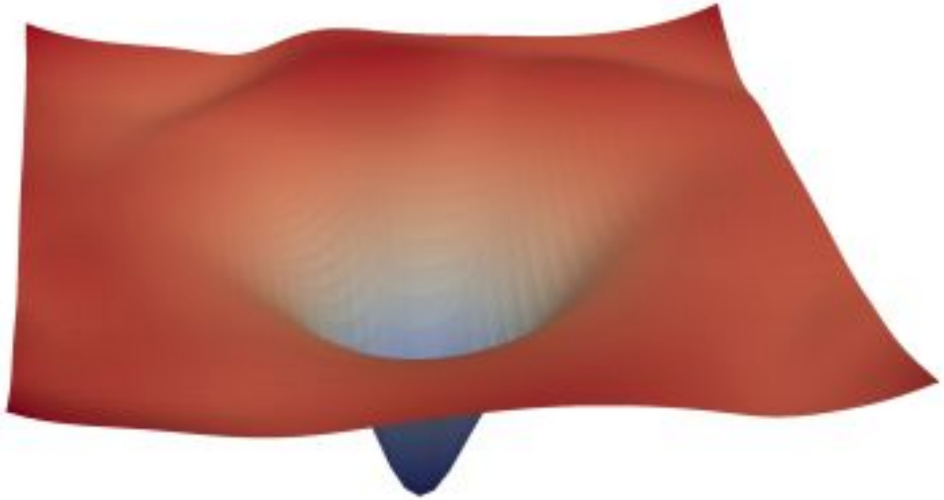
Training Data

Source of image:
http://arxiv.org/abs/1512.03385

每一種架構的存在都有一個理由

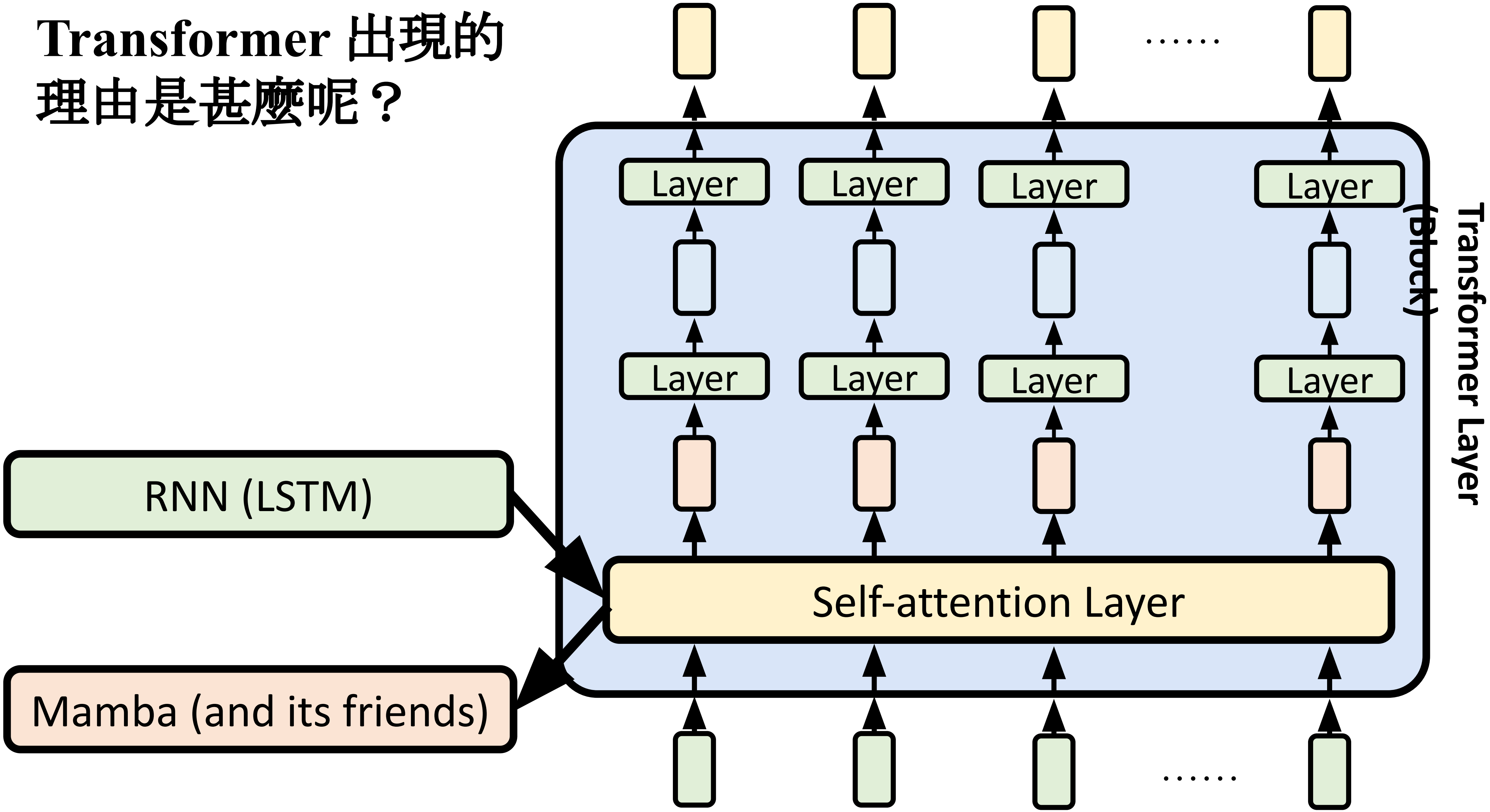
- Residual Connection 存在的理由是什麼？為了讓 Optimization 更容易


(a) without skip connections

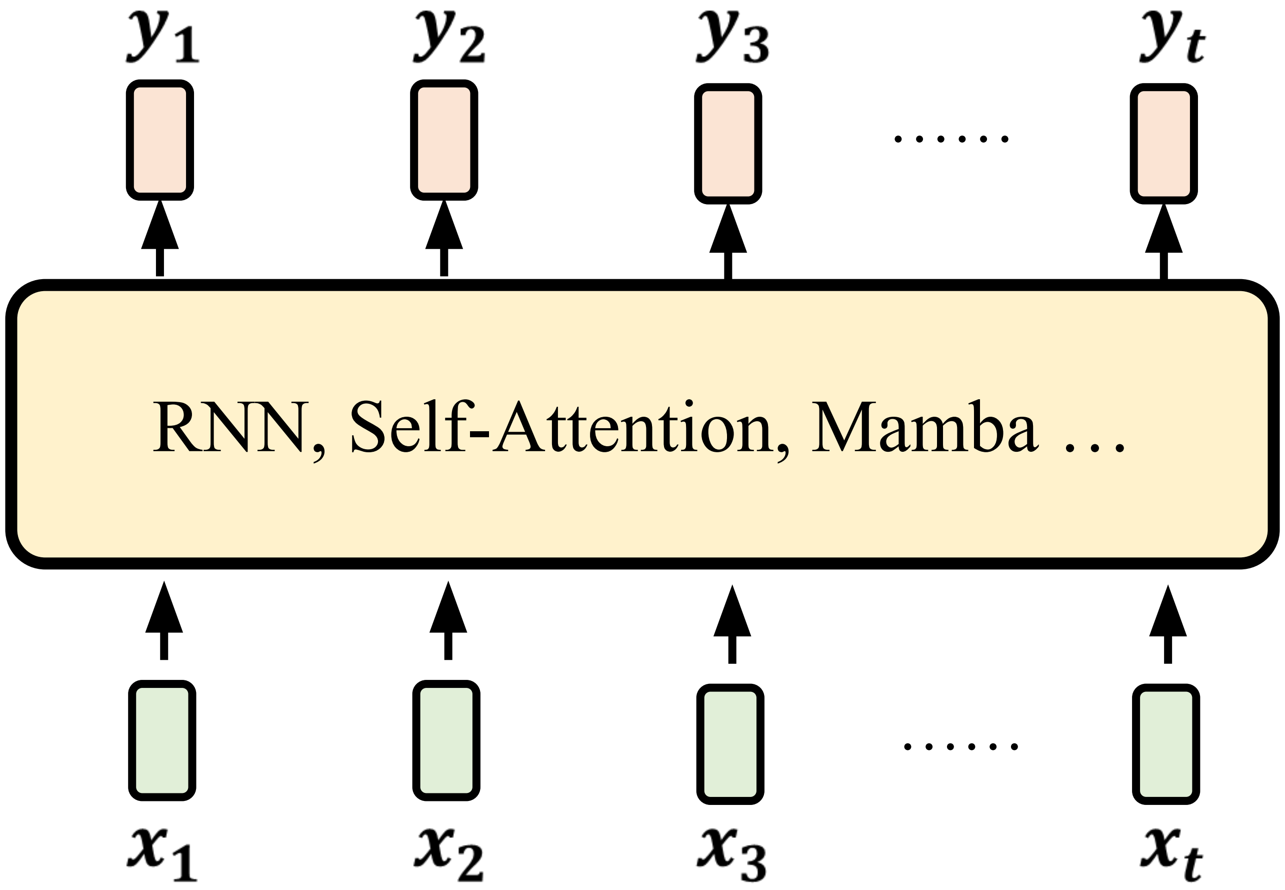

(b) with skip connections

Source of image: https://arxiv.org/abs/1712.09913

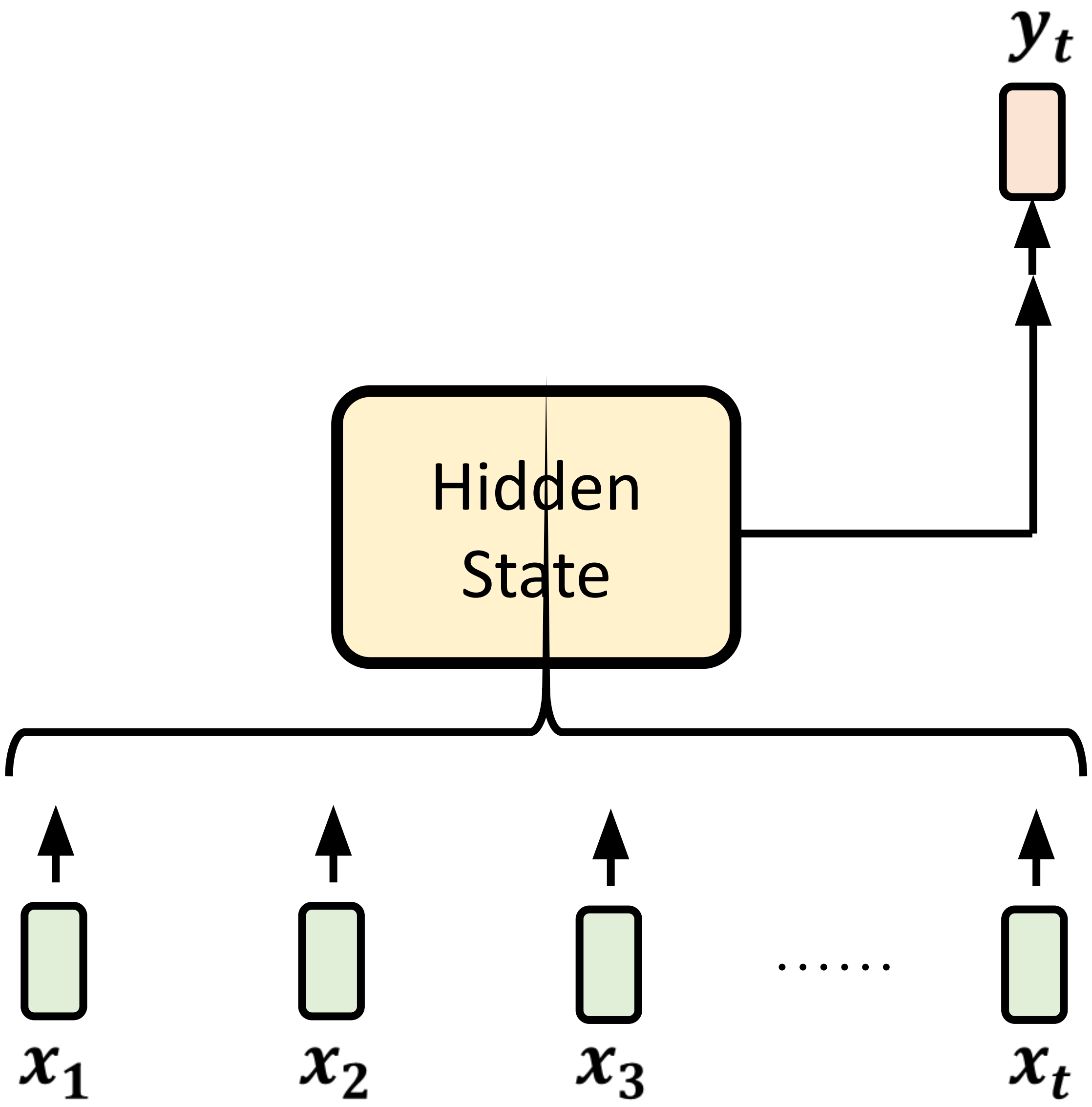
Transformer 出現的理由是甚麼呢？



要解的問題



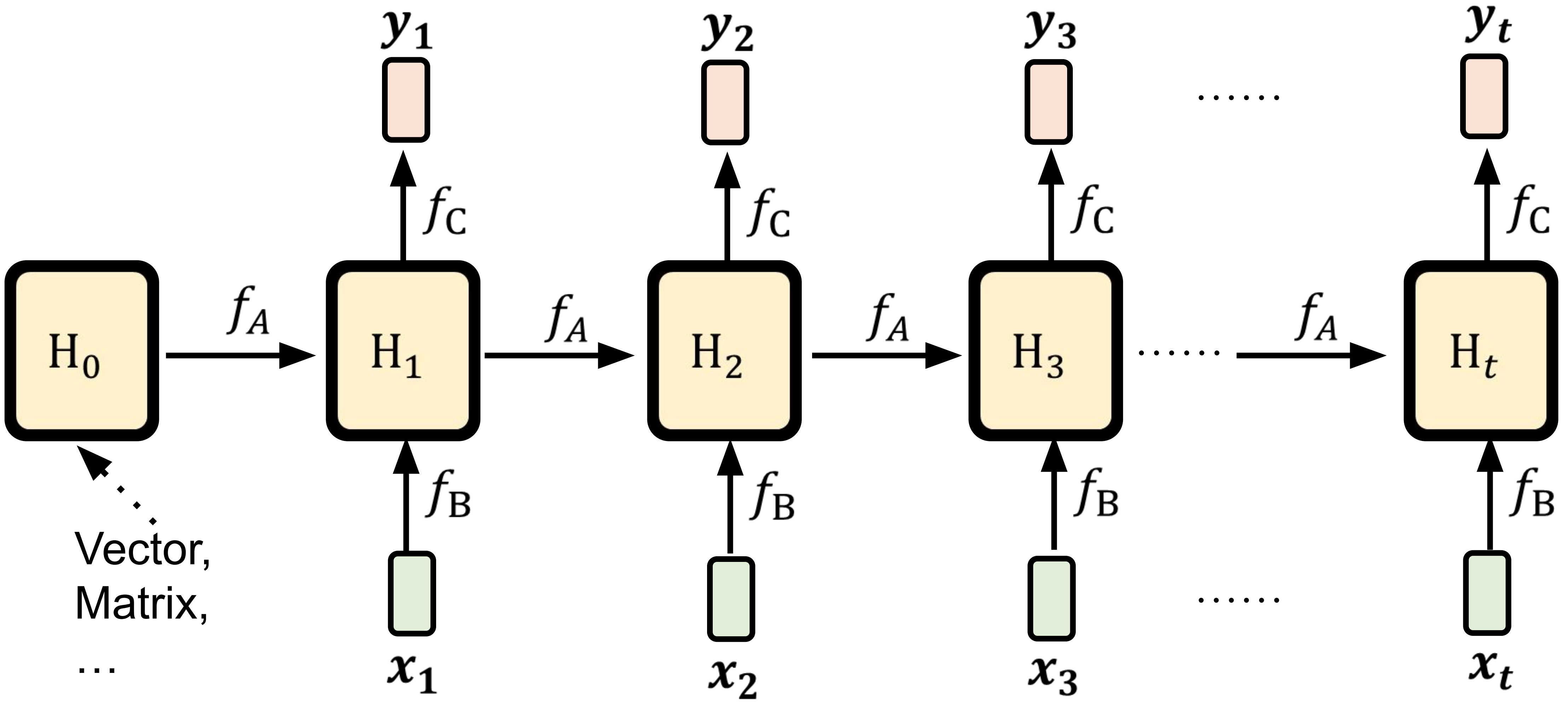
RNN-Style



RNN-Style

$$\mathrm{H}_t = f_A(\mathrm{H}_{t-1}) + f_B(\boldsymbol{x}_t)$$

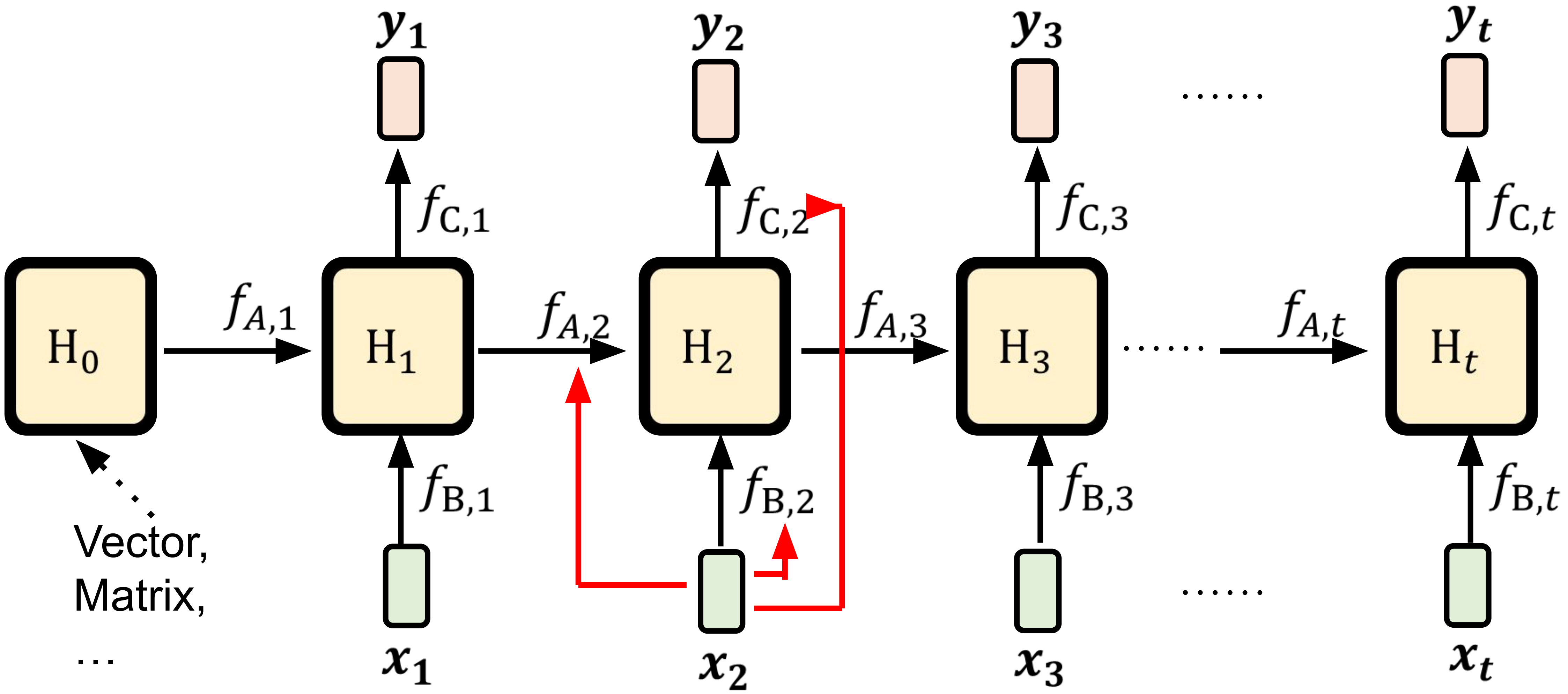
$$\boldsymbol{y}_t = f_C(\mathrm{H}_t)$$



RNN-Style

$$\mathrm{H}_t = f_{A,t}(\mathrm{H}_{t-1}) + f_{B,t}(\boldsymbol{x}_t)$$

$$\boldsymbol{y}_t = f_{C,t}(\mathrm{H}_t)$$

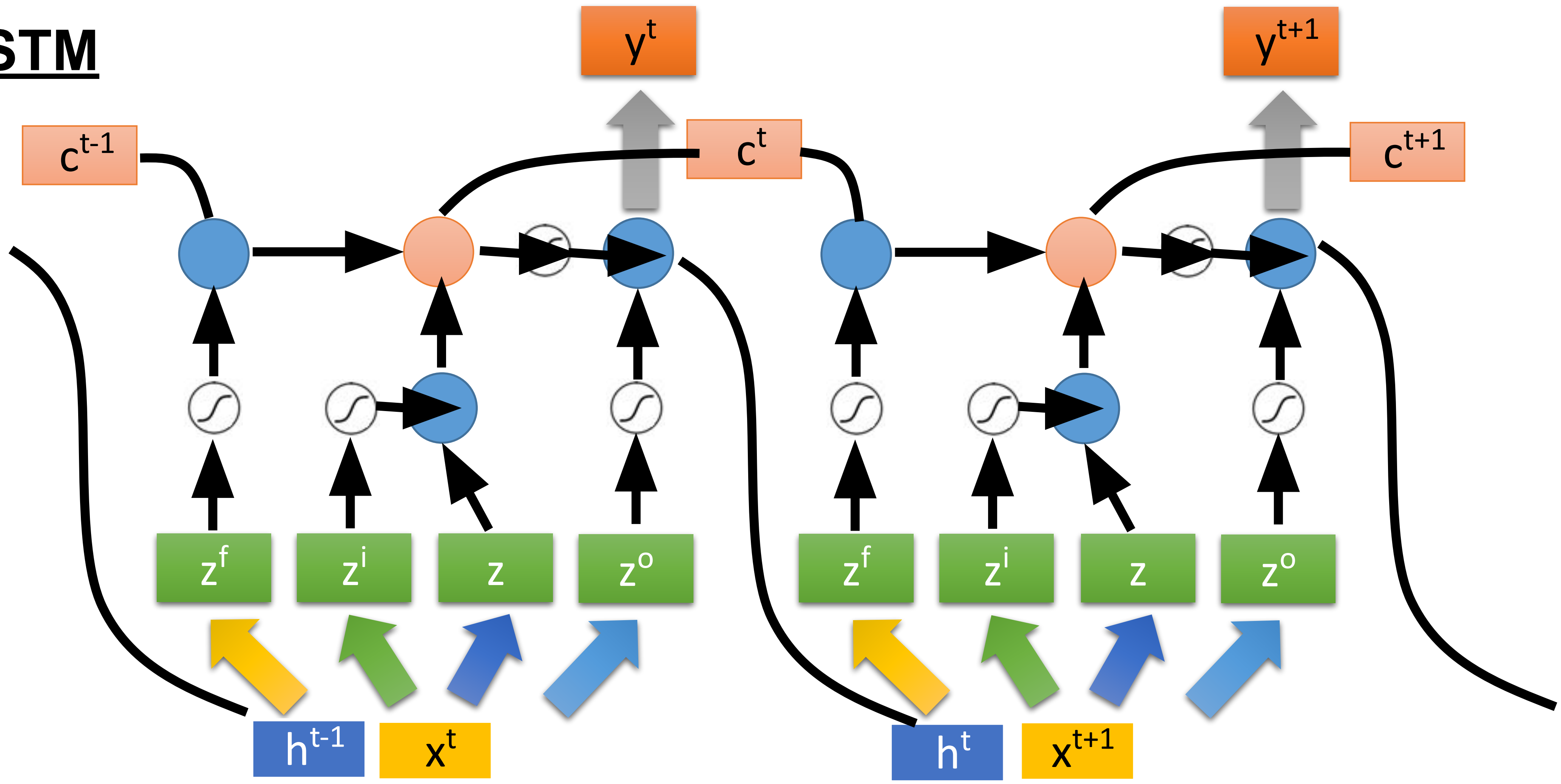


RNN-Style

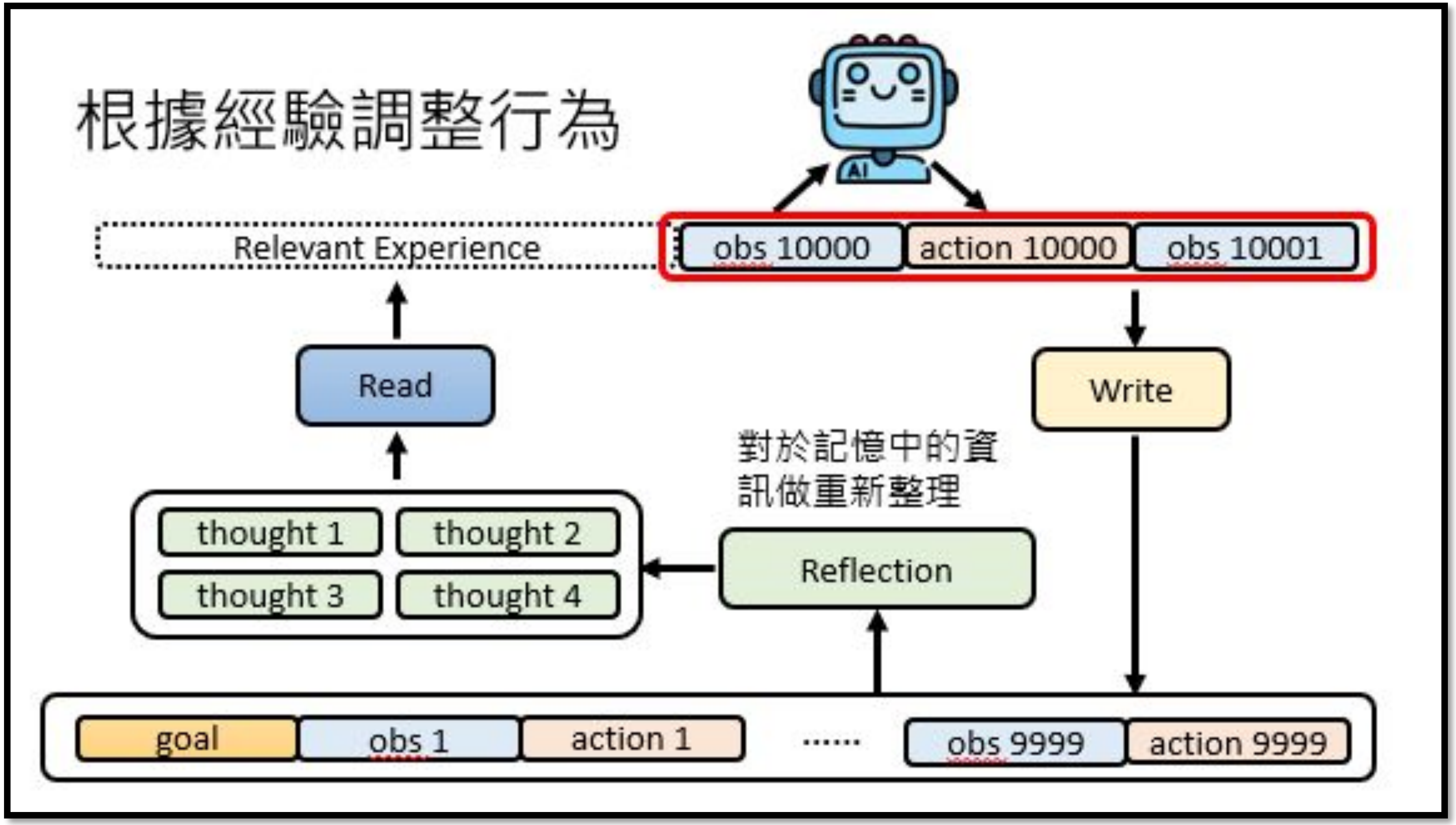
$$H_t = f_{A,t}(H_{t-1}) + f_{B,t}(\boldsymbol{x}_t)$$

$$\boldsymbol{y}_t = f_{C,t}(H_t)$$

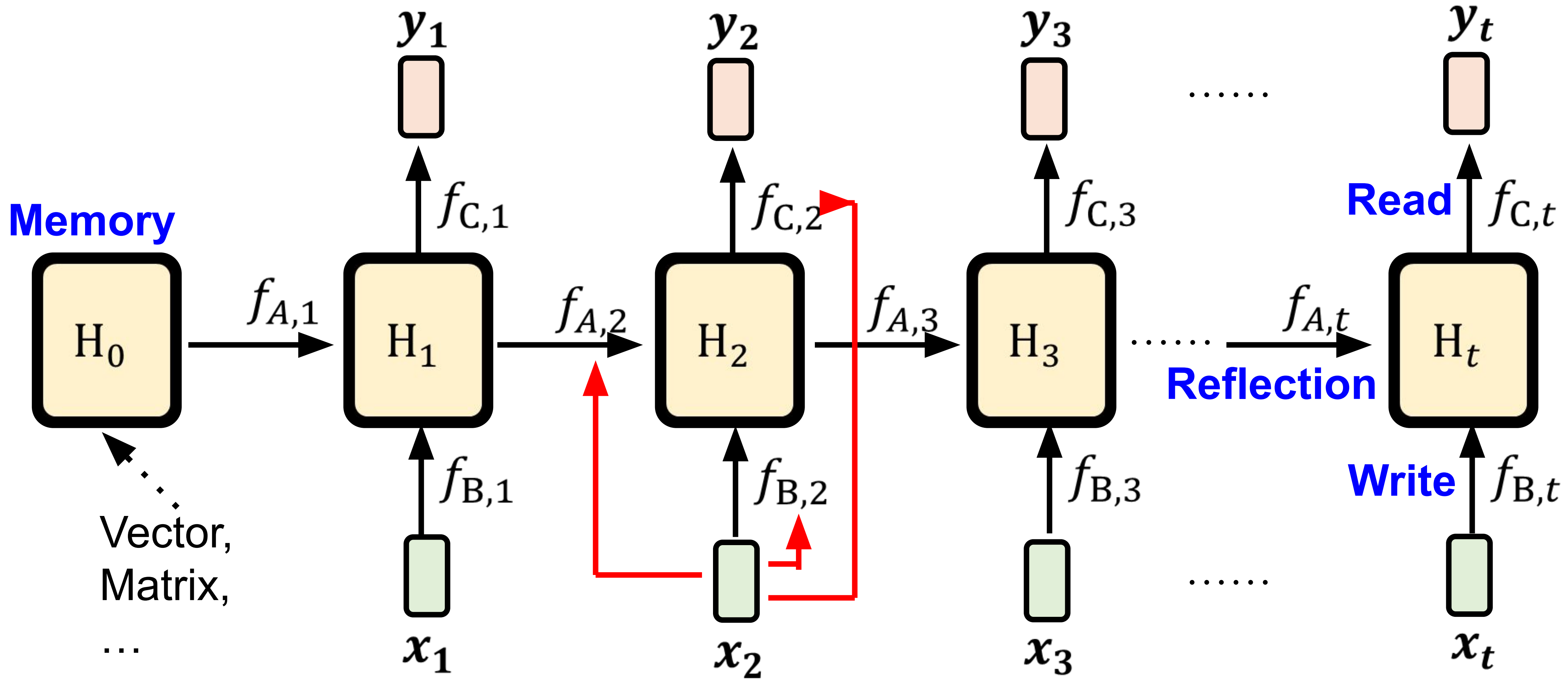
LSTM

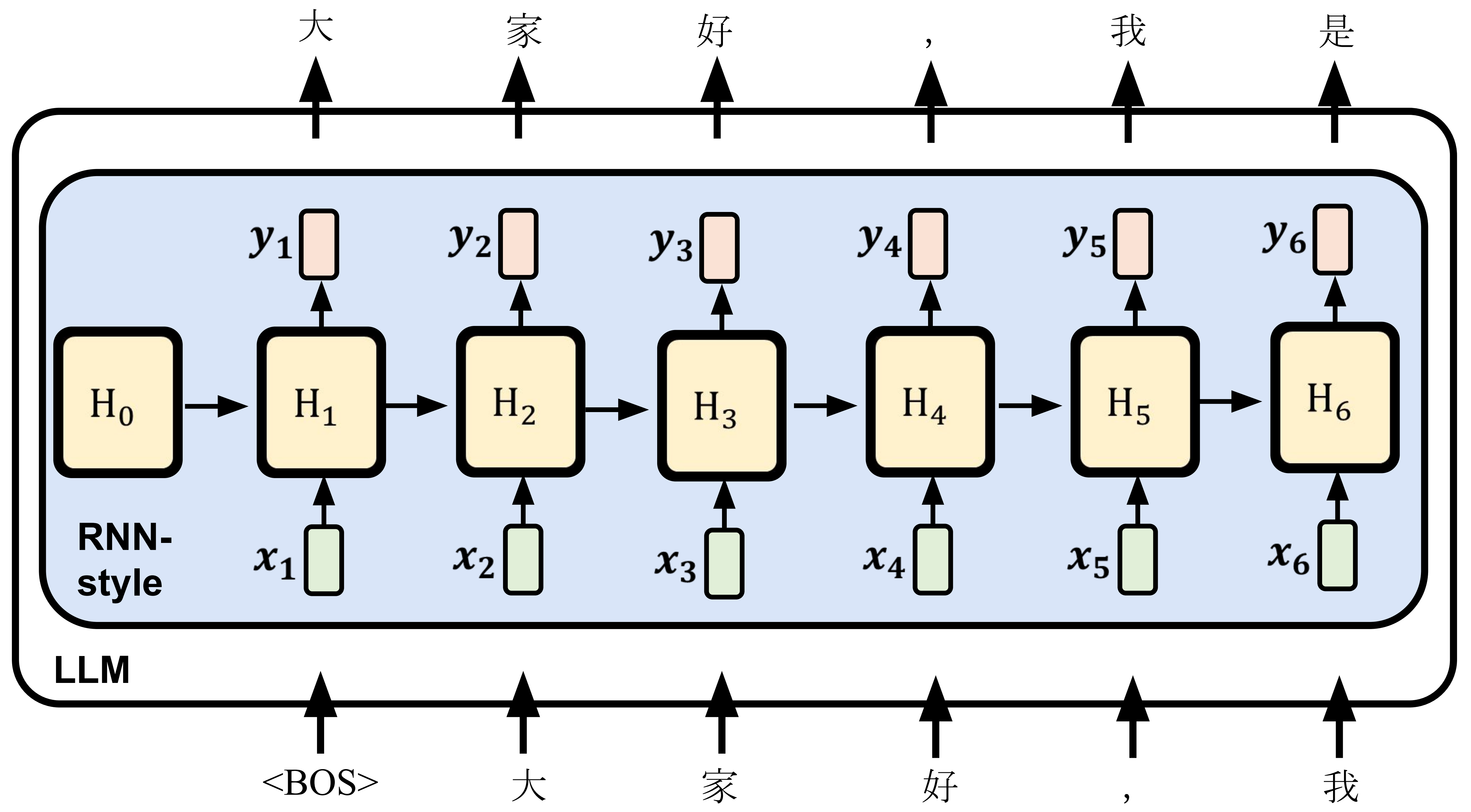


RNN-Style vs. AI Agent's Memory

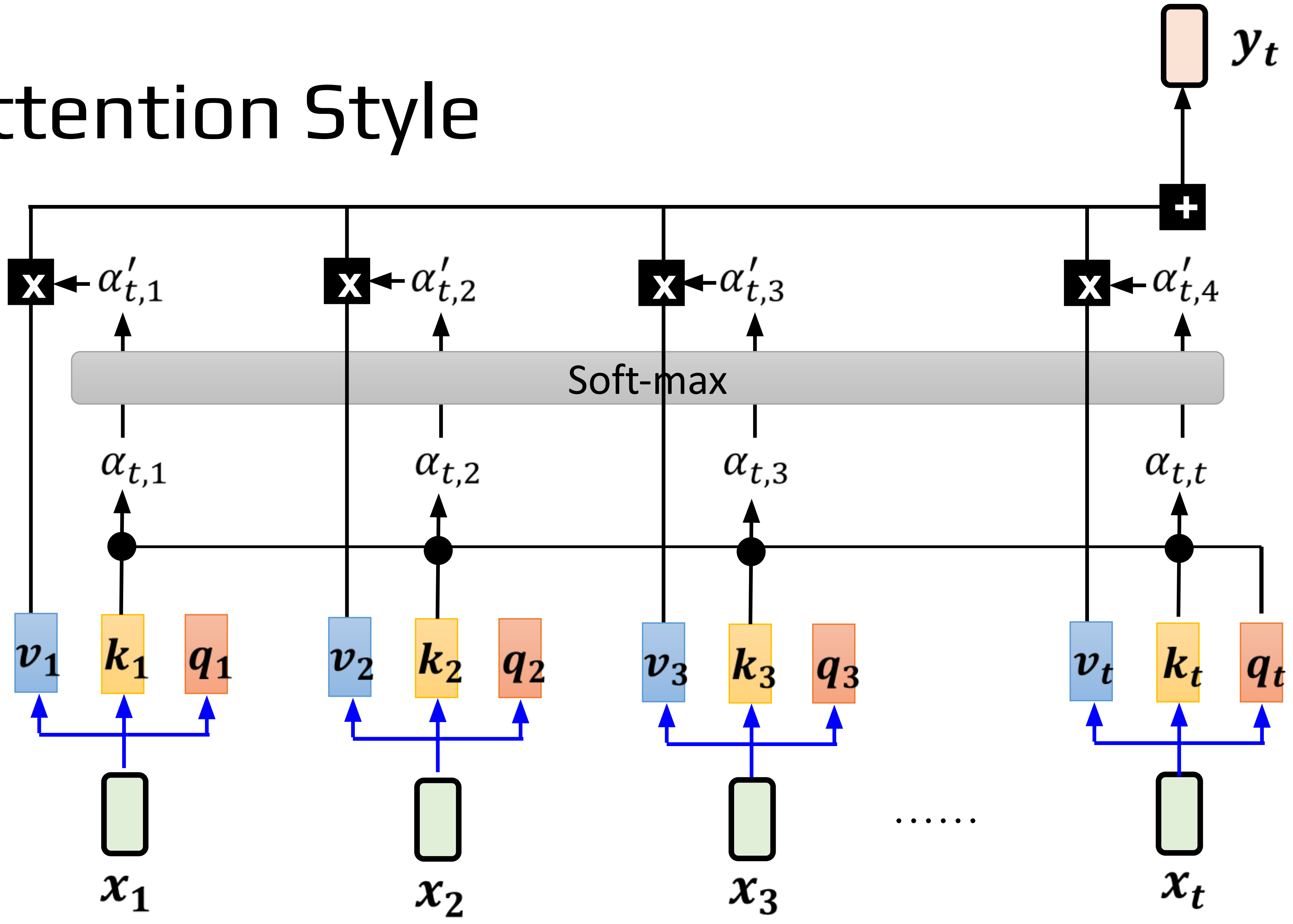


RNN-Style vs. AI Agent's Memory

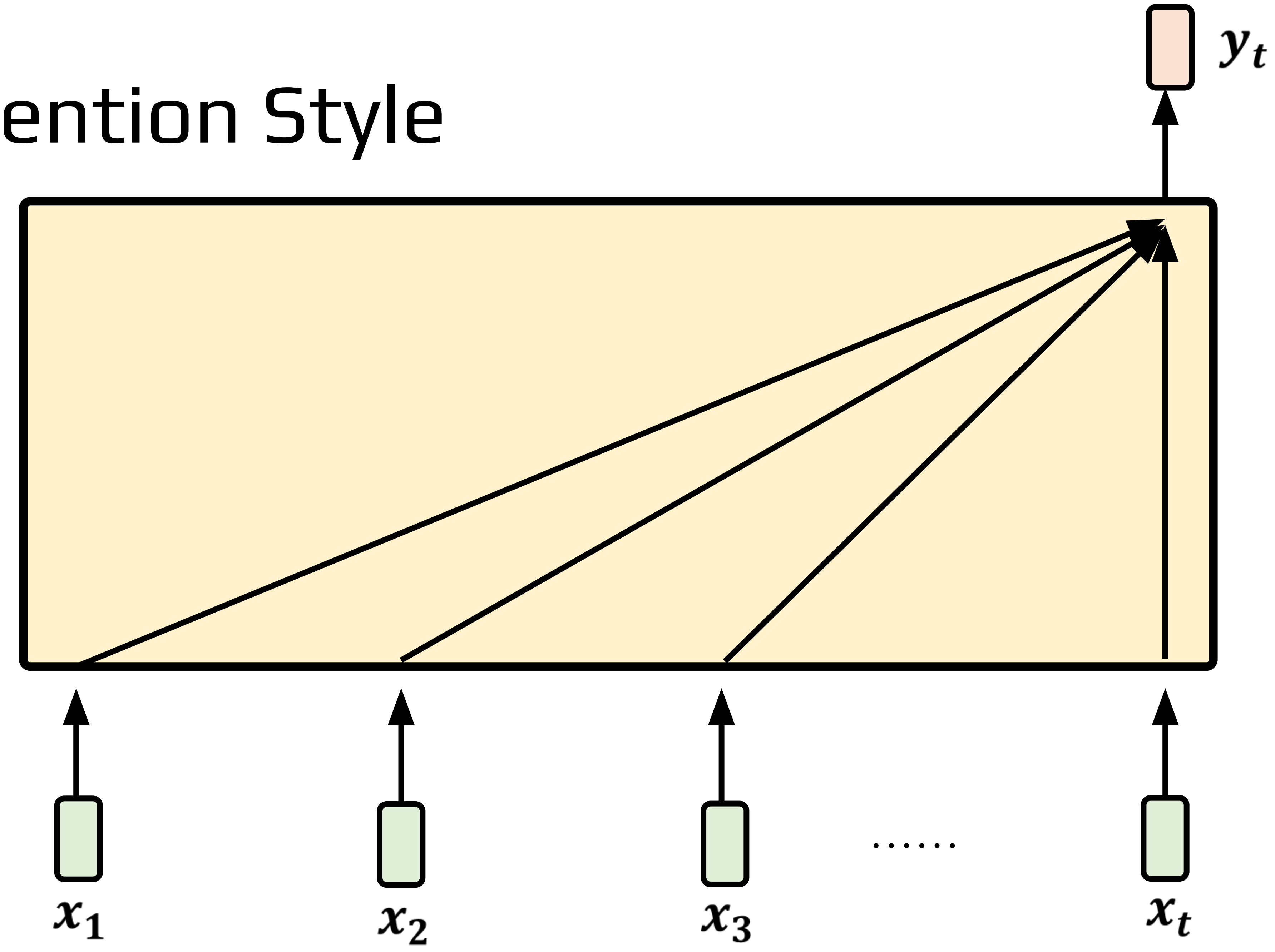



大
家
好
，
我
是
y_1
y_2
y_3
y_4
y_5
y_6
H_0
H_1
H_2
H_3
H_4
H_5
H_6
RNN-style
x_1
x_2
x_3
x_4
x_5
x_6
LLM
<BOS>
大
家
好
，
我

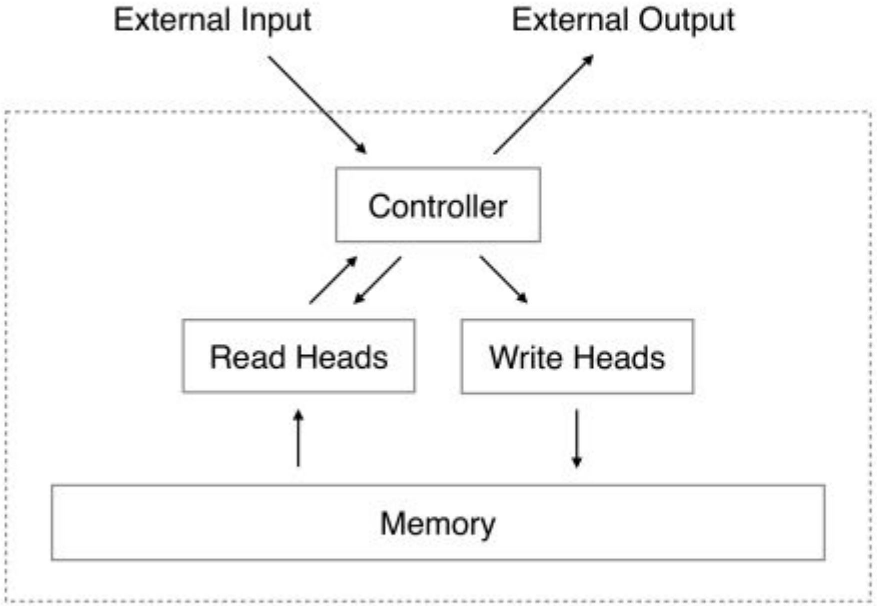
Self-Attention Style



Self-Attention Style



Attention 的概念很早就有了



Neural Turing Machine

https://arxiv.org/abs/1410.5401

Joe went to the garden then Fred picked up the milk; Joe moved to the bathroom and Fred dropped the milk, and then Dan moved to the living_room.
Where is Dan? A: living room I believe
Where is Joe? A: the bathroom
Fred moved to the bedroom and Joe went to the kitchen then Joe took the milk there and Dan journeyed to the bedroom; Joe discarded the milk.
Where is the milk now ? A: the milk is in the kitchen
Where is Dan now? A: I think he is in the bedroom
Joe took the milk there, after that Mike travelled to the office, then Joe went to the living_room, next Dan went back to the kitchen and Joe travelled to the office.
Where is Joe now? A: I think Joe is in the office

Memory Networks

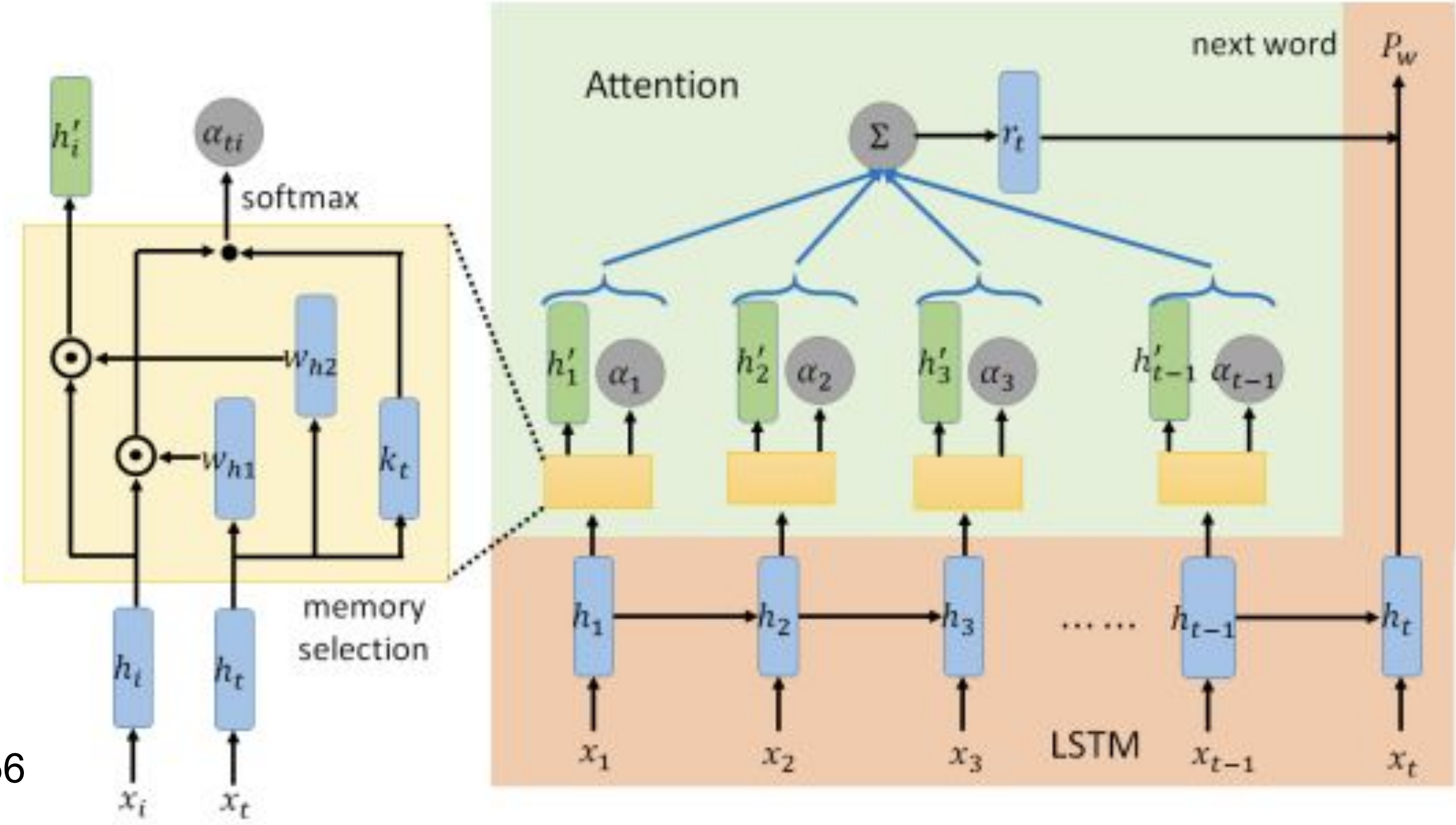
https://arxiv.org/pdf/1410.3916

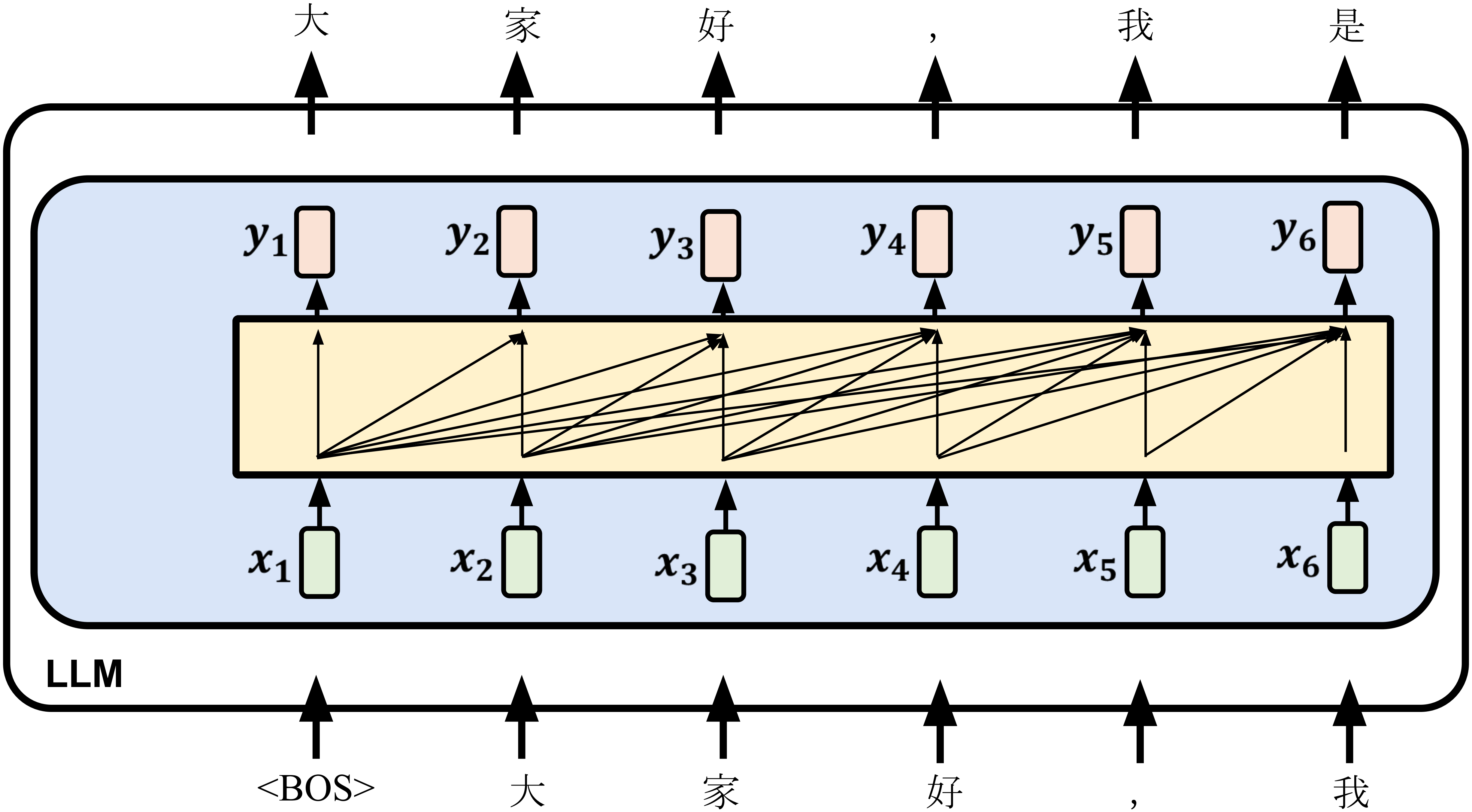
Attention 的概念很早就有了

Da-Rong Liu

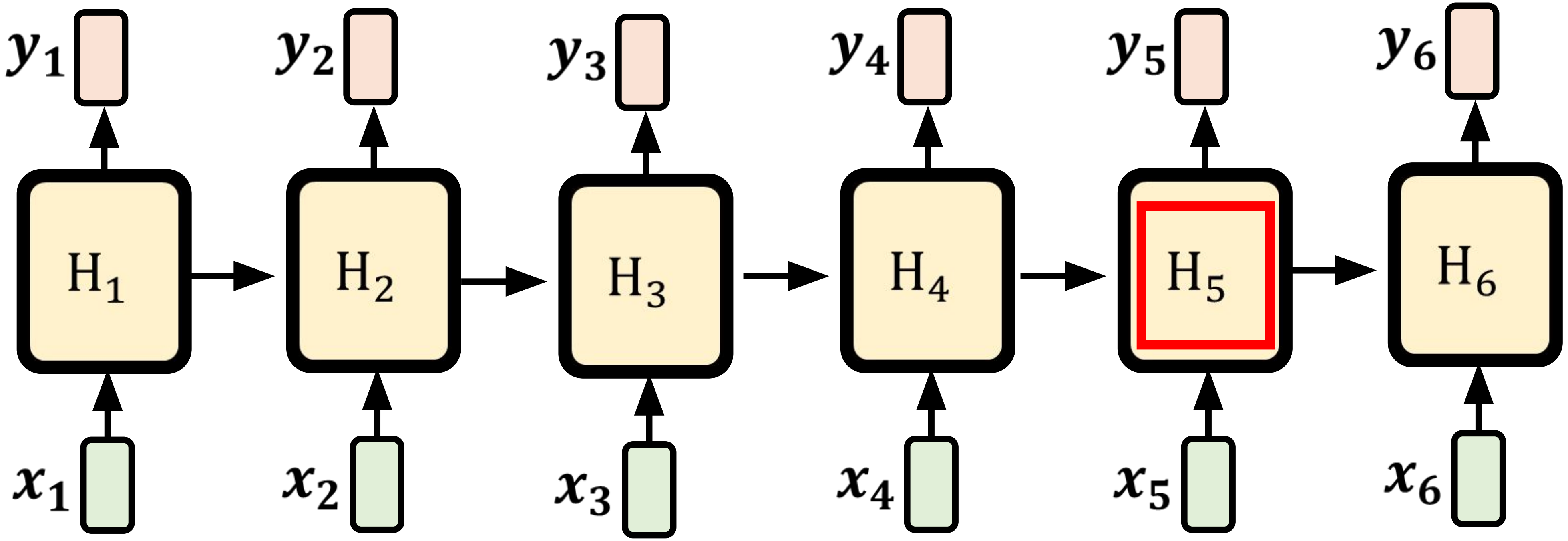
Attention-based Memory Selection Recurrent Network for Language Modeling

https://arxiv.org/abs/1611.08656



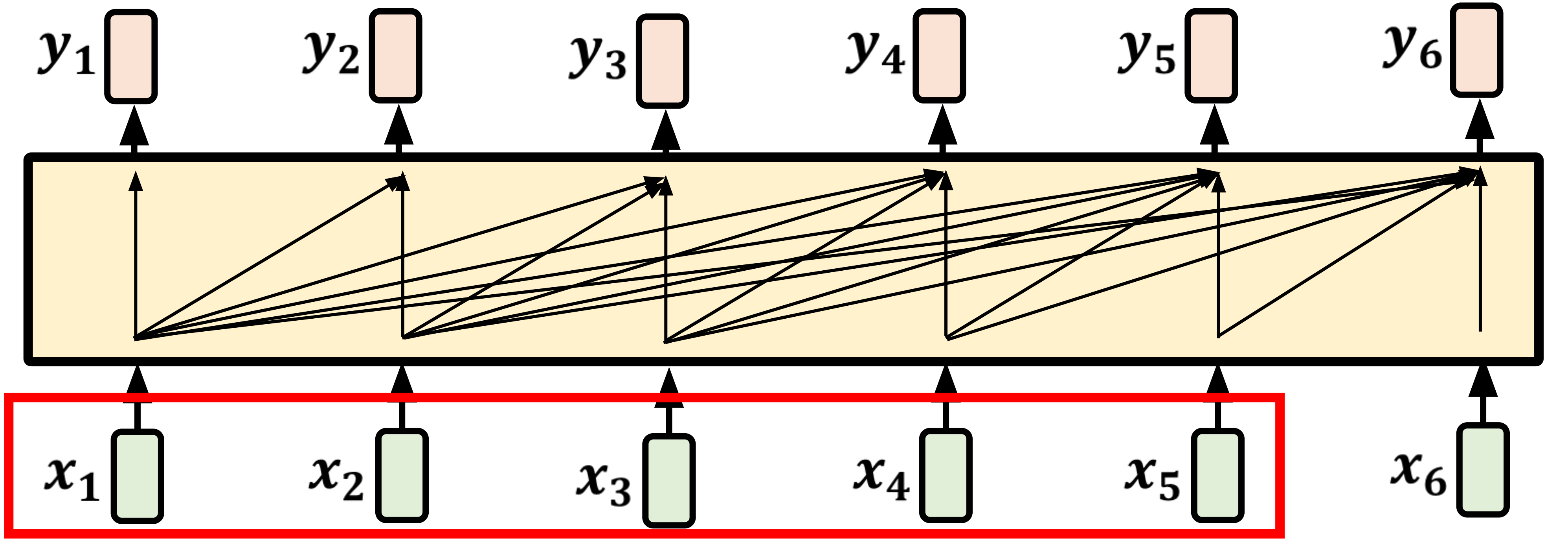

大
家
好
，
我
是
y_1
y_2
y_3
y_4
y_5
y_6
x_1
x_2
x_3
x_4
x_5
x_6
LLM
<BOS>
大
家
好
，
我

每一步運算量
都一樣



RNN 沒辦法記
大量資訊？

輸入越長，運
算量越來越大



Attention Is All You Need

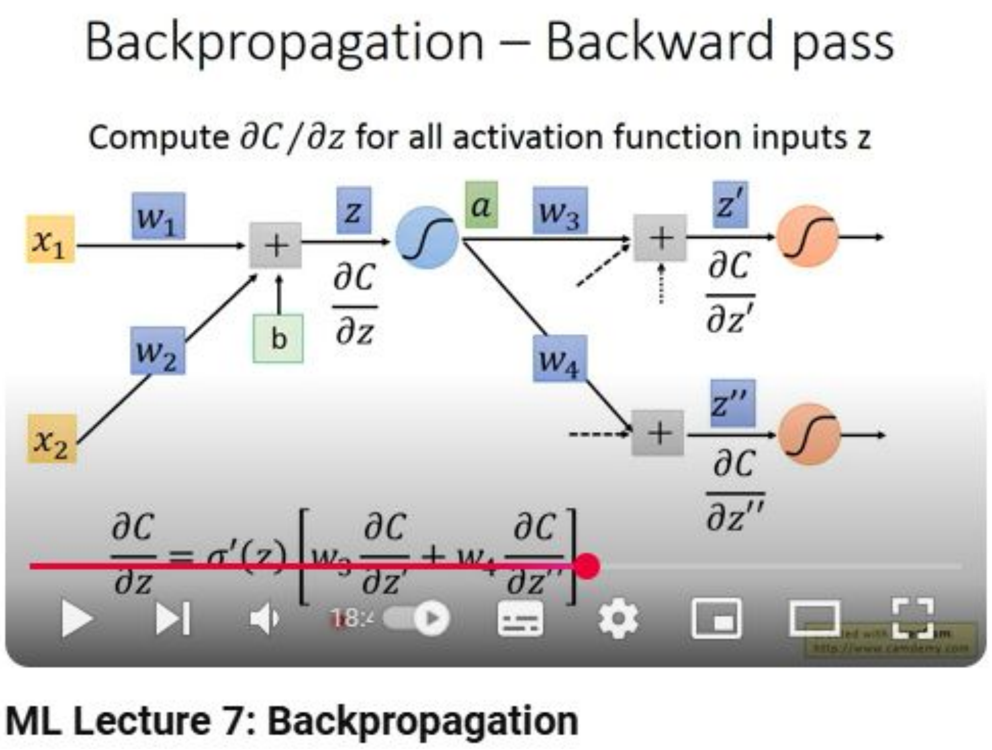
不是發明 Attention，而是拿掉 Attention 以外的東西

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

In this work we propose the Transformer, a model architecture eschewing recurrence and instead relying entirely on an attention mechanism to draw global dependencies between input and output. The Transformer allows for significantly more parallelization and can reach a new state of the art in translation quality after being trained for as little as twelve hours on eight P100 GPUs.

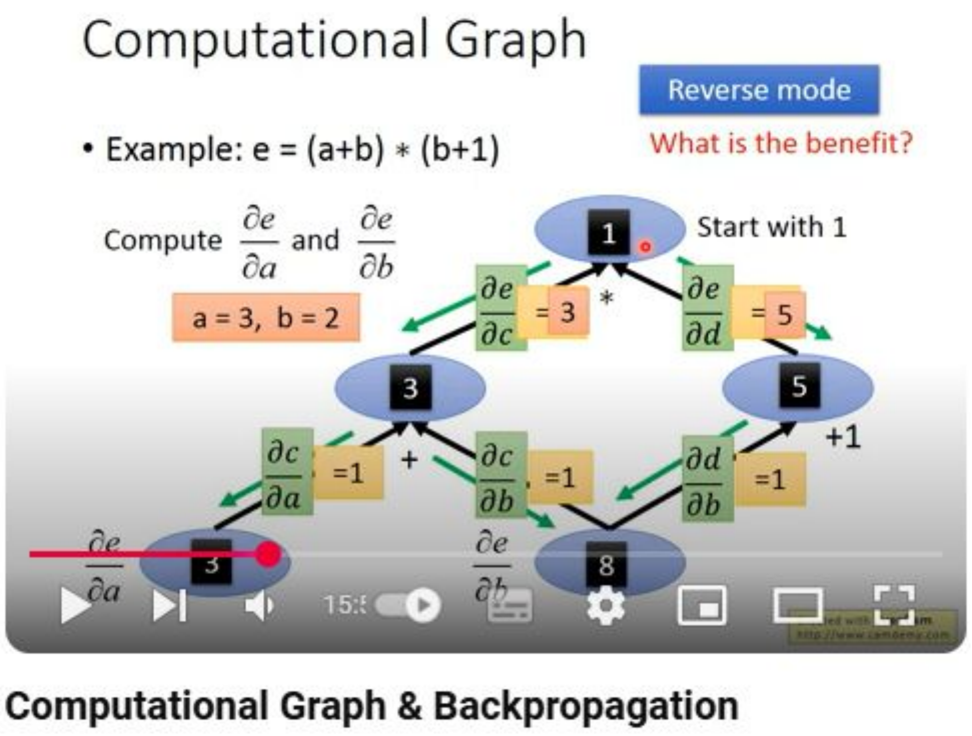
https://arxiv.org/abs/1706.03762

語言模型的訓練 (找出參數)



Backpropagation

https://youtu.be/ibJpTrp5mcE



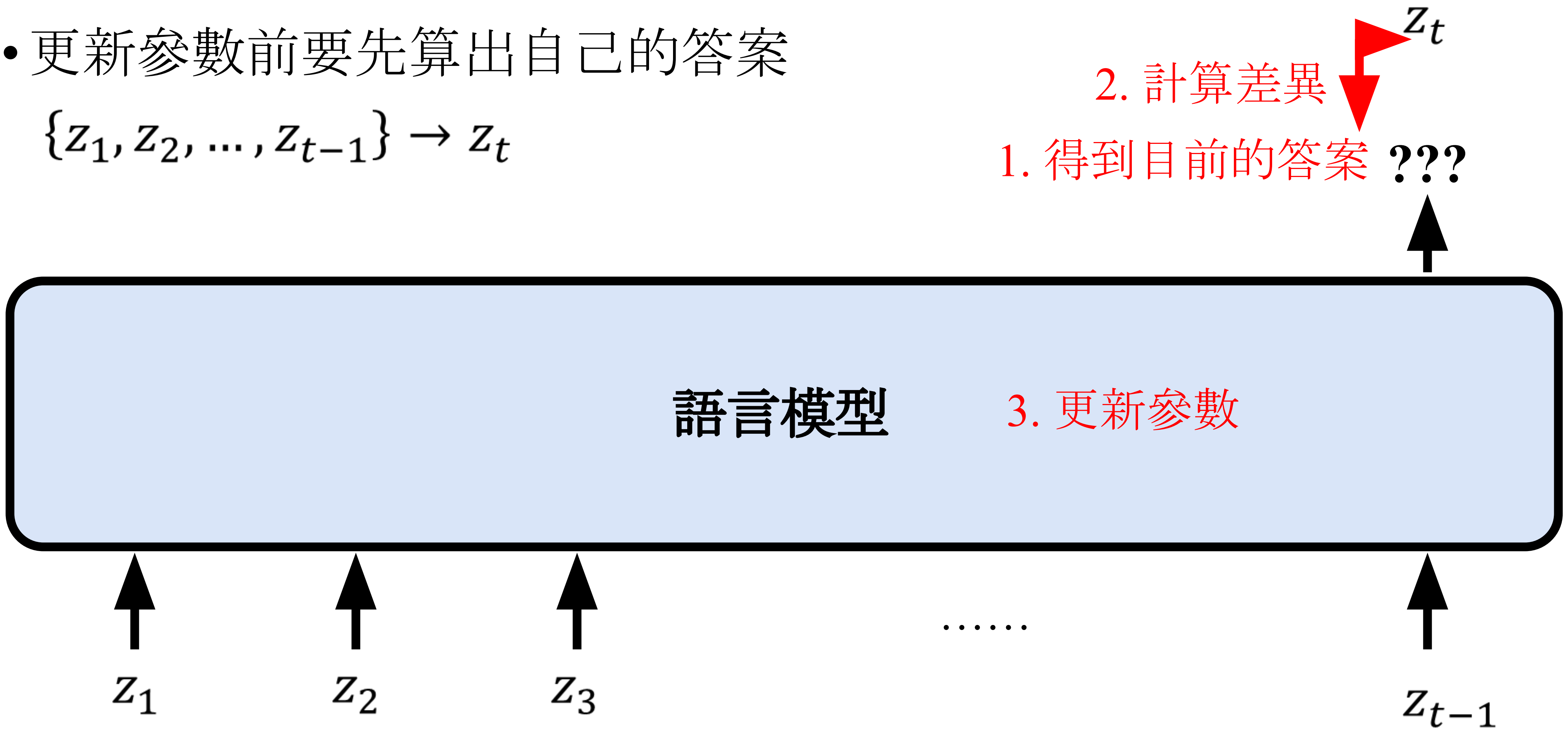
Computational Graph

https://youtu.be/-yhm3WdGFok?si=2cZOANbtm0Mjd9IT

語言模型的訓練 (找出參數)

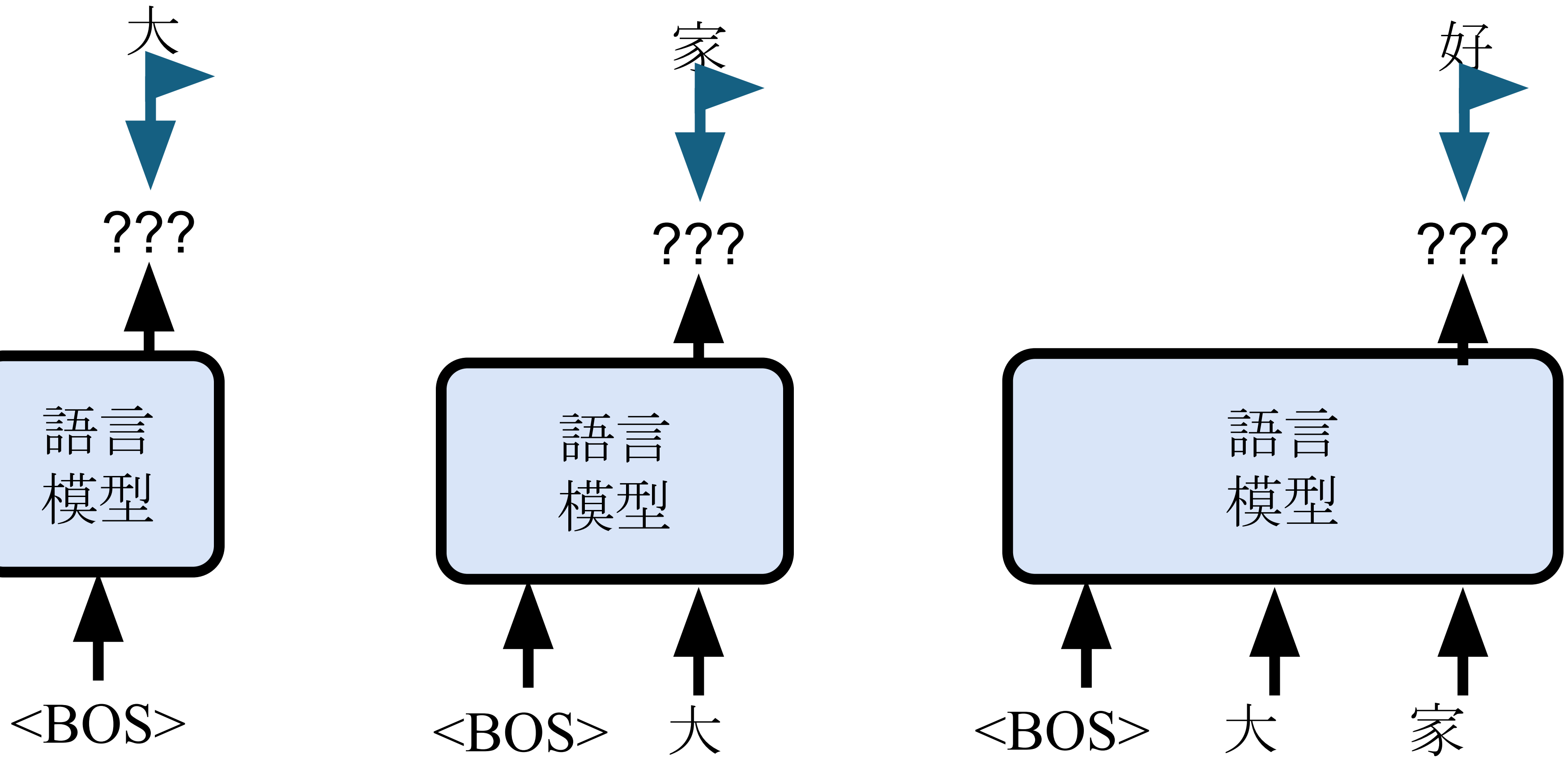
- 更新參數前要先算出自己的答案

$\{z_1, z_2, \dots, z_{t-1}\} \rightarrow z_t$



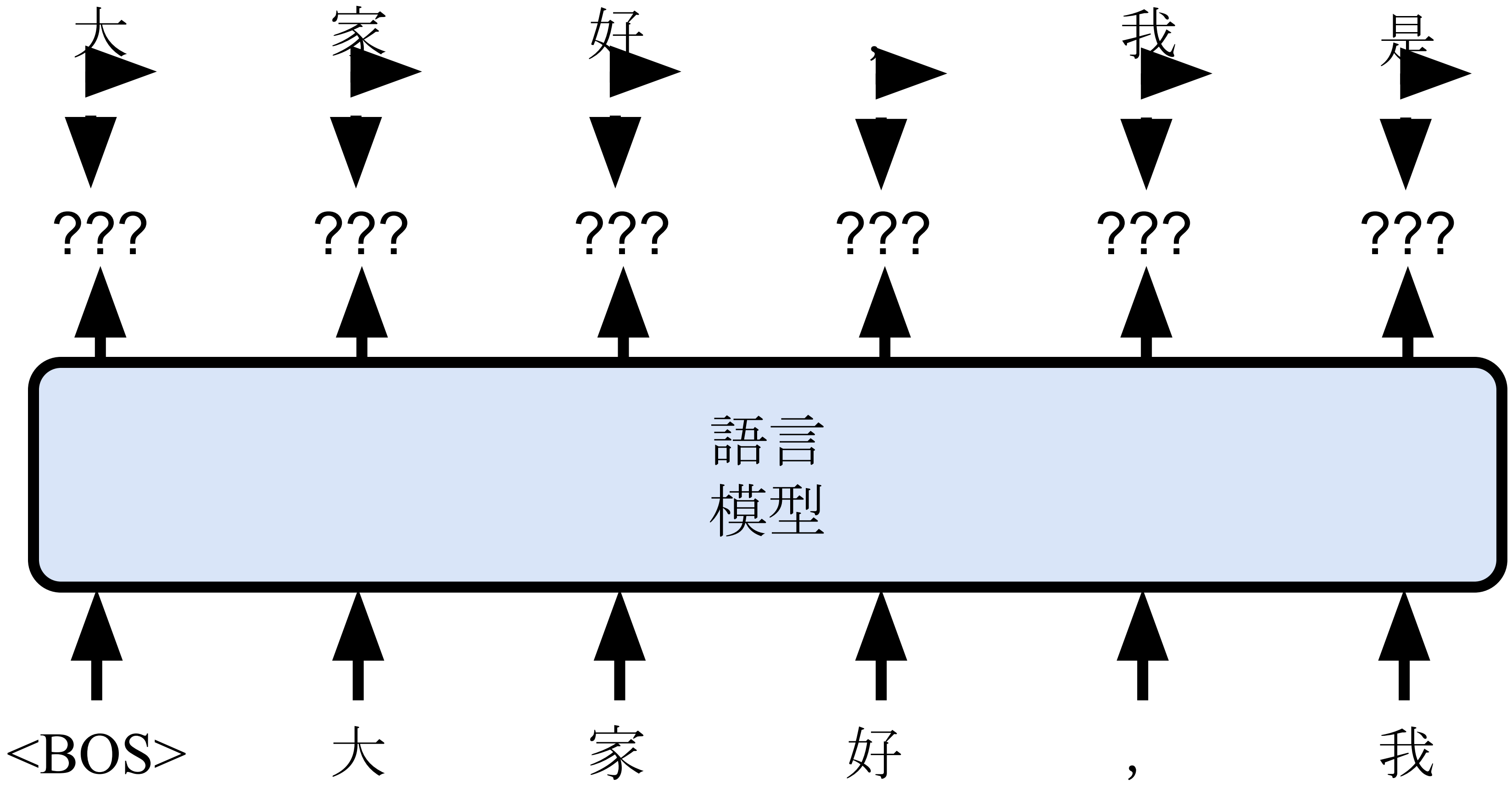
語言模型的訓練 (找出參數)

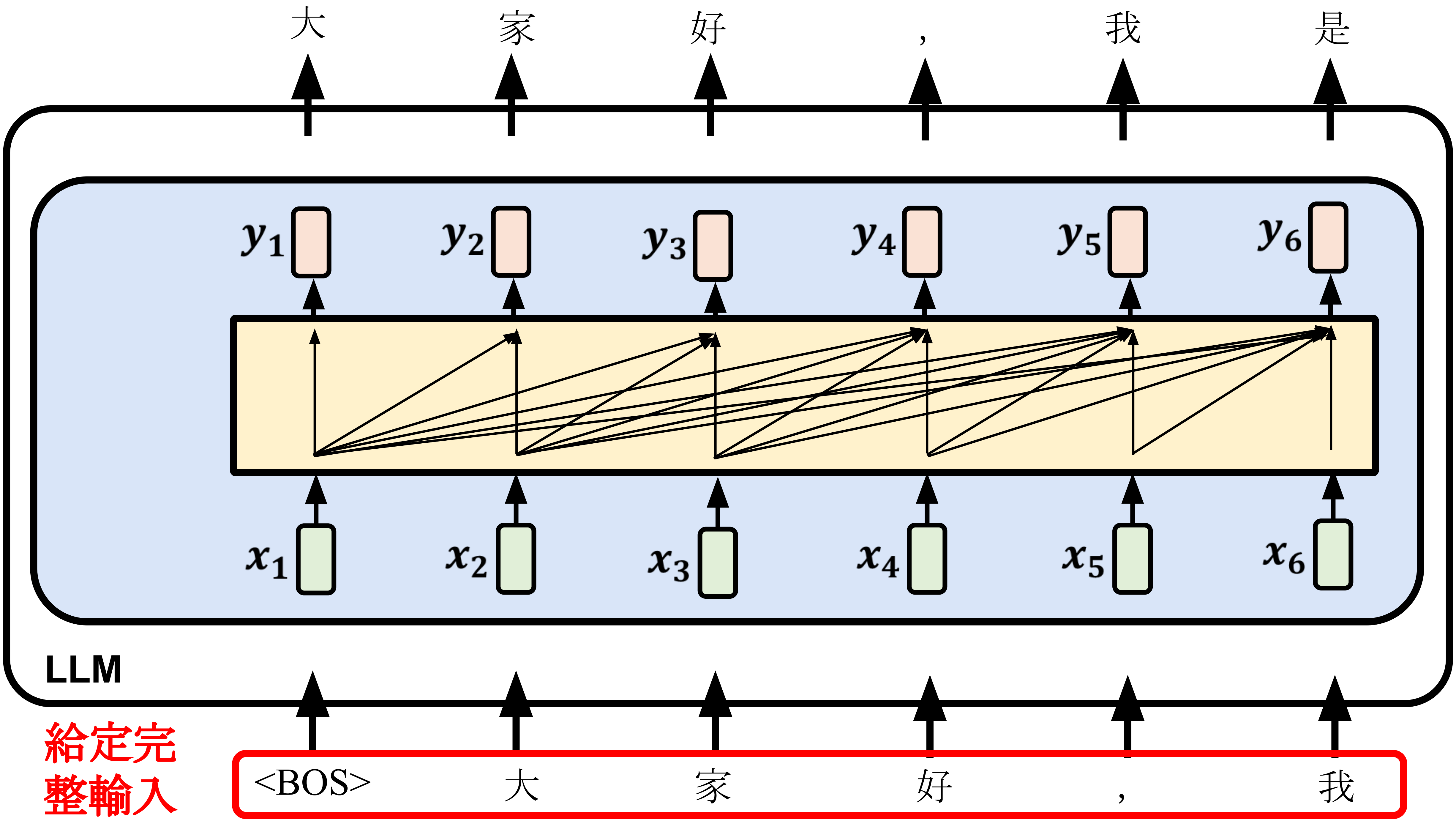
假設我們想要教模型說「大 家 好 ， 我 是 ……」

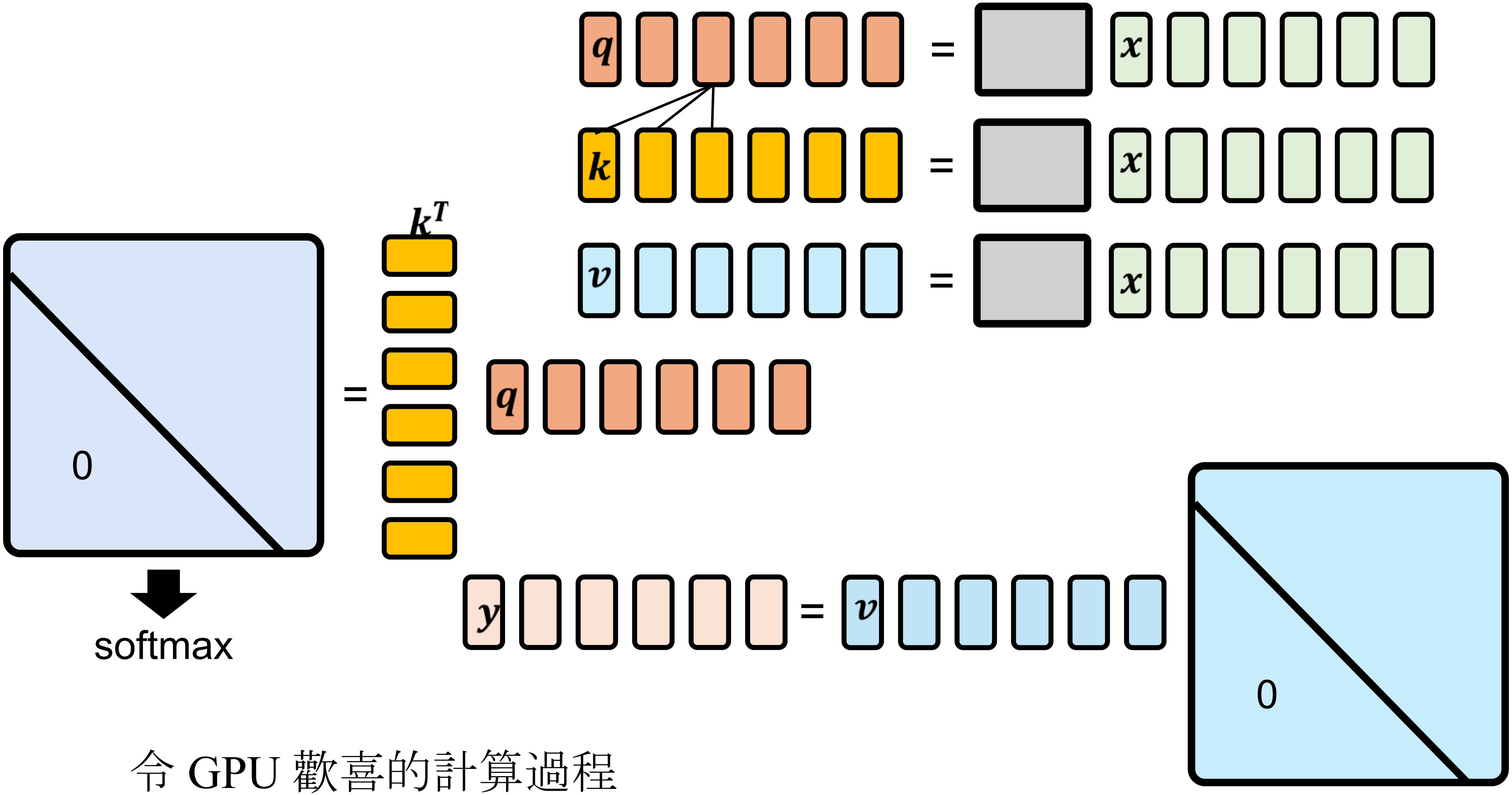


語言模型的訓練 (找出參數)

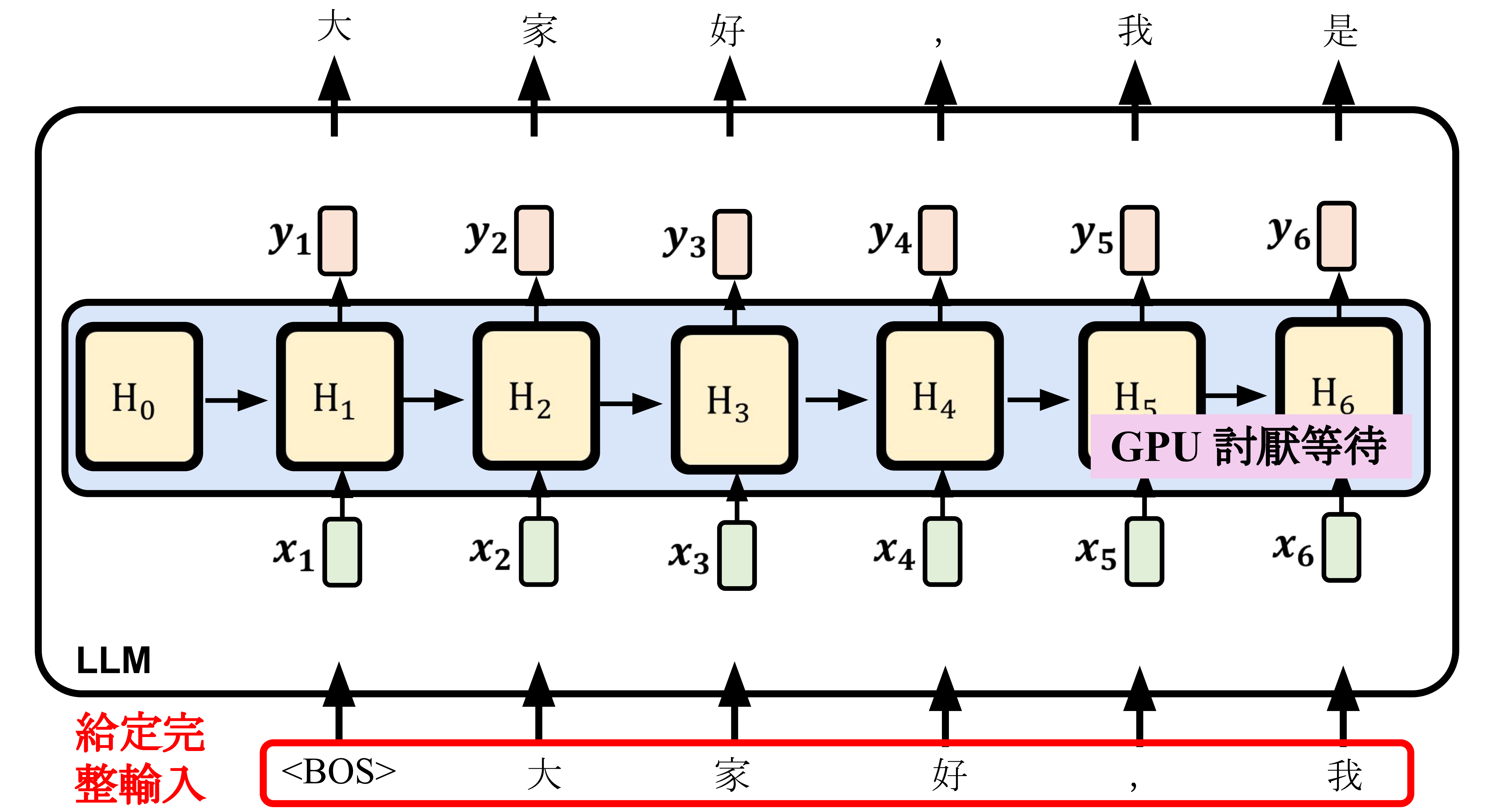
假設我們想要教模型說「大　家　好　，　我　是 ……」




大 家 好 ， 我 是
y_1 y_2 y_3 y_4 y_5 y_6
x_1 x_2 x_3 x_4 x_5 x_6
LLM
給定完整輸入
<BOS> 大 家 好 ， 我



令 GPU 歡喜的計算過程



大
家
好
，
我
是
y_1
y_2
y_3
y_4
y_5
y_6
H_0
H_1
H_2
H_3
H_4
H_5
H_6
GPU 討厭等待
x_1
x_2
x_3
x_4
x_5
x_6
LLM
給定完整輸入
<BOS>
大
家
好
，
我

Self-attention vs. RNN-style

	Self-attention	**RNN**
Inference	計算量、記憶體需求隨著序列長度增加	計算量、記憶體需求固定
Training	容易平行化	難以平行化(？)

RAG、AI Agent 都需要語言模型處理很長的序列

影像、聲音是比文字更長的序列

Source of image: https://www.artfish.ai/p/long-context-llms

RNN 有沒有訓練時平行的可能性

$$H_t = f_{A,t}(H_{t-1}) + f_{B,t}(\boldsymbol{x_t})$$

$$\boldsymbol{y_t} = f_{C,t}(H_t)$$

$$f_{A,1}(H_0) = O$$

$$\begin{cases} H_1 = f_{A,1}(H_0) + f_{B,1}(\boldsymbol{x_1}) & = f_{B,1}(\boldsymbol{x_1}) \\ H_2 = f_{A,2}(H_1) + f_{B,2}(\boldsymbol{x_2}) & = f_{A,2}\left(f_{B,1}(\boldsymbol{x_1})\right) + f_{B,2}(\boldsymbol{x_2}) \\ H_3 = f_{A,3}(H_2) + f_{B,3}(\boldsymbol{x_3}) & = f_{A,3}\left(f_{A,2}\left(f_{B,1}(\boldsymbol{x_1})\right) + f_{B,2}(\boldsymbol{x_2})\right) + f_{B,3}(\boldsymbol{x_3}) \\ \vdots & \\ H_t = f_{A,t}(H_{t-1}) + f_{B,t}(\boldsymbol{x_t}) & = \underbrace{f_{A,t}(f_{A,t-1} \dots f_{A,3}(f_{A,2}}(f_{B,1}(\boldsymbol{x_1}) \dots) \quad \dots + f_{B,t}(\boldsymbol{x_t}) \end{cases}$$

RNN 有沒有訓練時平行的可能性

$$\mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x_t})$$

$$\boldsymbol{y_t} = f_{C,t}(\mathrm{H}_t)$$

$f_{A,1}(\mathrm{H}_0) = O$

$$\begin{cases} \mathrm{H}_1 = \mathrm{H}_0 + f_{B,1}(\boldsymbol{x_1}) & = f_{B,1}(\boldsymbol{x_1}) \\ \mathrm{H}_2 = \mathrm{H}_1 + f_{B,2}(\boldsymbol{x_2}) & = f_{B,1}(\boldsymbol{x_1}) + f_{B,2}(\boldsymbol{x_2}) \\ \mathrm{H}_3 = \mathrm{H}_2 + f_{B,3}(\boldsymbol{x_3}) & = f_{B,1}(\boldsymbol{x_1}) + f_{B,2}(\boldsymbol{x_2}) + f_{B,3}(\boldsymbol{x_3}) \\ \vdots & \\ \mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x_t}) & = f_{B,1}(\boldsymbol{x_1}) + f_{B,2}(\boldsymbol{x_2}) + f_{B,3}(\boldsymbol{x_3}) \quad \ldots\ldots + f_{B,t}(\boldsymbol{x_t}) \end{cases}$$

H_t is a $d \times d$ matric

$f_{B,t}(\boldsymbol{x_t}) = D_t$

RNN 有沒有訓練時平行的可能性

$$\mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x_t})$$

$$f_{A,1}(\mathrm{H}_0) = O$$

$$\boldsymbol{y_t} = f_{C,t}(\mathrm{H}_t)$$

$$\begin{cases} \mathrm{H}_1 = D_1 & \boldsymbol{y_1} = D_1\boldsymbol{q_1} \\ \mathrm{H}_2 = D_1 + D_2 & \boldsymbol{y_2} = D_1\boldsymbol{q_2} + D_2\boldsymbol{q_2} \\ \mathrm{H}_3 = D_1 + D_2 + D_3 & \boldsymbol{y_3} = D_1\boldsymbol{q_3} + D_2\boldsymbol{q_3} + D_3\boldsymbol{q_3} \\ \vdots & \\ \mathrm{H}_t = D_1 + D_2 + \cdots + D_t & \boldsymbol{y_t} = D_1\boldsymbol{q_t} + D_2\boldsymbol{q_t} + \cdots + D_t\boldsymbol{q_t} \end{cases}$$

H_t is a $d \times d$ matric

$$f_{B,t}(\boldsymbol{x_t}) = D_t$$

$$f_{C,t}(\mathrm{H}_t) = \mathrm{H}_t\boldsymbol{q_t}$$

$$\boldsymbol{q_t} = W_Q\boldsymbol{x_t}$$

RNN 有沒有訓練時平行的可能性

$$f_{A,1}(\mathrm{H}_0) = O$$

$$\begin{cases} \boldsymbol{y_1} = D_1\boldsymbol{q_1} \\ \boldsymbol{y_2} = D_1\boldsymbol{q_2} + D_2\boldsymbol{q_2} \\ \boldsymbol{y_3} = D_1\boldsymbol{q_3} + D_2\boldsymbol{q_3} + D_3\boldsymbol{q_3} \\ \vdots \\ \boldsymbol{y_t} = D_1\boldsymbol{q_t} + D_2\boldsymbol{q_t} + \cdots + D_t\boldsymbol{q_t} \end{cases}$$

$$\mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x_t})$$

$$\boldsymbol{y_t} = f_{C,t}(\mathrm{H}_t)$$

H_t is a $d \times d$ matric

$$f_{B,t}(\boldsymbol{x_t}) = D_t$$

$$D_t = \boldsymbol{v_t}\boldsymbol{k_t}^T \quad \boldsymbol{v_t} = W_v\boldsymbol{x_t} \quad \boldsymbol{k_t} = W_k\boldsymbol{x_t}$$

$$f_{C,t}(\mathrm{H}_t) = \mathrm{H}_t\boldsymbol{q_t}$$

$$\boldsymbol{q_t} = W_Q\boldsymbol{x_t}$$

RNN 有沒有訓練時平行的可能性

$f_{A,1}(\mathrm{H}_0) = O$

$$\begin{cases} \boldsymbol{y_1} = \boldsymbol{v_1}\boldsymbol{k_1}^T\boldsymbol{q_1} \\ \boldsymbol{y_2} = \boldsymbol{v_1}\boldsymbol{k_1}^T\boldsymbol{q_2} + \boldsymbol{v_2}\boldsymbol{k_2}^T\boldsymbol{q_2} \\ \boldsymbol{y_3} = \boldsymbol{v_1}\boldsymbol{k_1}^T\boldsymbol{q_3} + \boldsymbol{v_2}\boldsymbol{k_2}^T\boldsymbol{q_3} + \boldsymbol{v_3}\boldsymbol{k_3}^T\boldsymbol{q_3} \\ \vdots \\ \boldsymbol{y_t} = \boldsymbol{v_1}\boldsymbol{k_1}^T\boldsymbol{q_t} + \boldsymbol{v_2}\boldsymbol{k_2}^T\boldsymbol{q_t} + \cdots + \boldsymbol{v_t}\boldsymbol{k_t}^T\boldsymbol{q_t} \end{cases}$$

$$\mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x_t})$$

$$\boldsymbol{y_t} = f_{C,t}(\mathrm{H}_t)$$

H_t is a $d \times d$ matric

$$f_{B,t}(\boldsymbol{x_t}) = D_t$$

$$D_t = \boldsymbol{v_t}\boldsymbol{k_t}^T \qquad \boldsymbol{v_t} = W_v\boldsymbol{x_t} \qquad \boldsymbol{k_t} = W_k\boldsymbol{x_t}$$

$$f_{C,t}(\mathrm{H}_t) = \mathrm{H}_t\boldsymbol{q_t}$$

$$\boldsymbol{q_t} = W_Q\boldsymbol{x_t}$$

RNN 有沒有訓練時平行的可能性

$$f_{A,1}(\mathrm{H}_0) = O$$

$$\boldsymbol{y_t} = \boldsymbol{v_1}\boldsymbol{k_1}^T\boldsymbol{q_t} + \boldsymbol{v_2}\boldsymbol{k_2}^T\boldsymbol{q_t} + \cdots + \boldsymbol{v_t}\boldsymbol{k_t}^T\boldsymbol{q_t}$$

$$= \boldsymbol{v_1}a_{t,1} + \boldsymbol{v_2}a_{t,2} + \cdots + \boldsymbol{v_t}a_{t,t}$$

$$= a_{t,1}\boldsymbol{v_1} + a_{t,2}\boldsymbol{v_2} + \cdots + a_{t,t}\boldsymbol{v_t}$$

這不就是 Self-attention!　(少了 softmax)

叫做 Linear Attention

$$\mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x_t})$$

$$\boldsymbol{y_t} = f_{C,t}(\mathrm{H}_t)$$

H_t is a $d \times d$ matric

$$f_{B,t}(\boldsymbol{x_t}) = D_t$$

$$D_t = \boldsymbol{v_t}\boldsymbol{k_t}^T$$

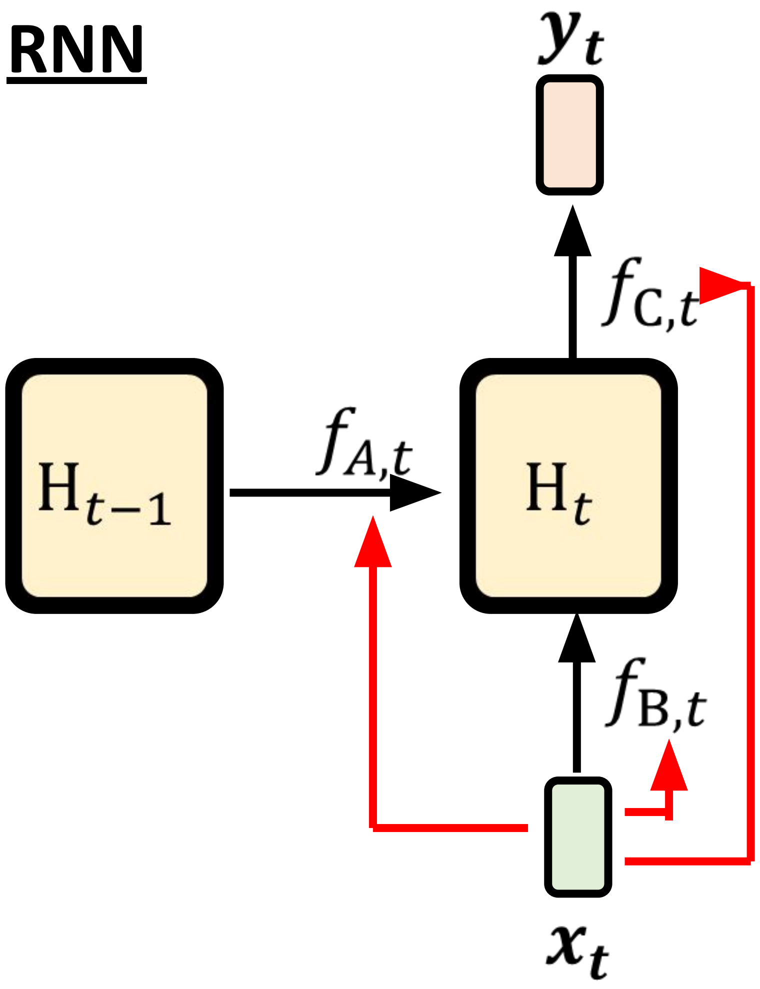
$$\boldsymbol{v_t} = W_v\boldsymbol{x_t}$$

$$\boldsymbol{k_t} = W_k\boldsymbol{x_t}$$

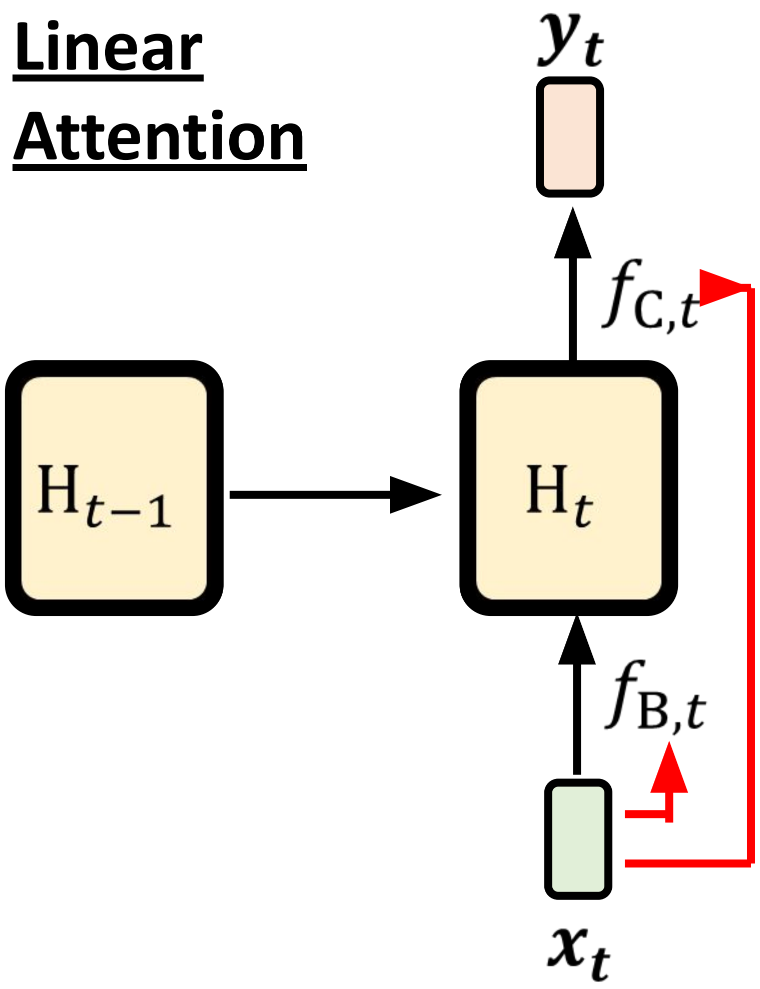
$$f_{C,t}(\mathrm{H}_t) = \mathrm{H}_t\boldsymbol{q_t}$$

$$\boldsymbol{q_t} = W_Q\boldsymbol{x_t}$$

RNN



Linear Attention



$$f_{C,t}(\mathrm{H}_t) = \mathrm{H}_t \boldsymbol{q}_t$$

$$\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$$

$$f_{B,t}(\boldsymbol{x}_t) = \boldsymbol{v}_t {\boldsymbol{k}_t}^T$$

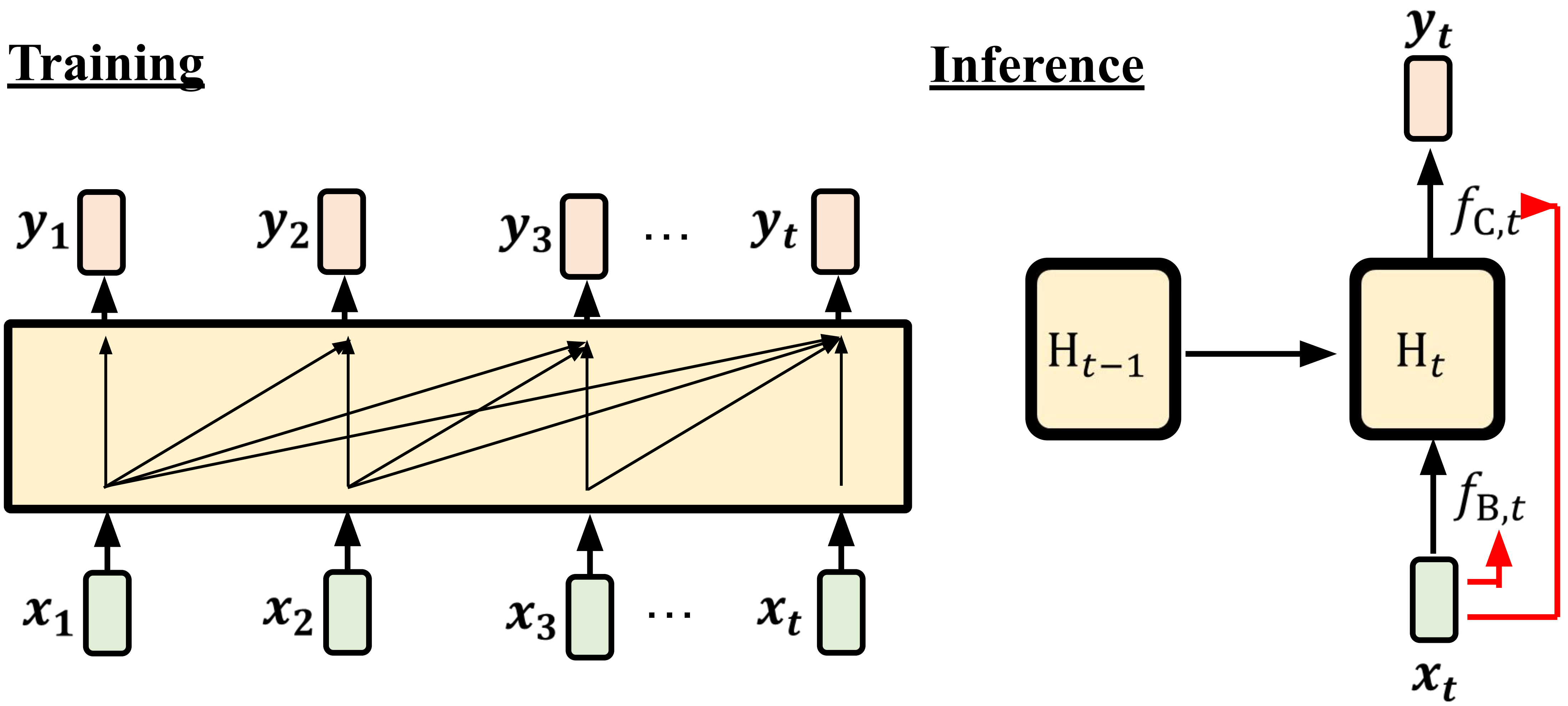
$$\boldsymbol{v}_t = W_v \boldsymbol{x}_t$$

$$\boldsymbol{k}_t = W_k \boldsymbol{x}_t$$

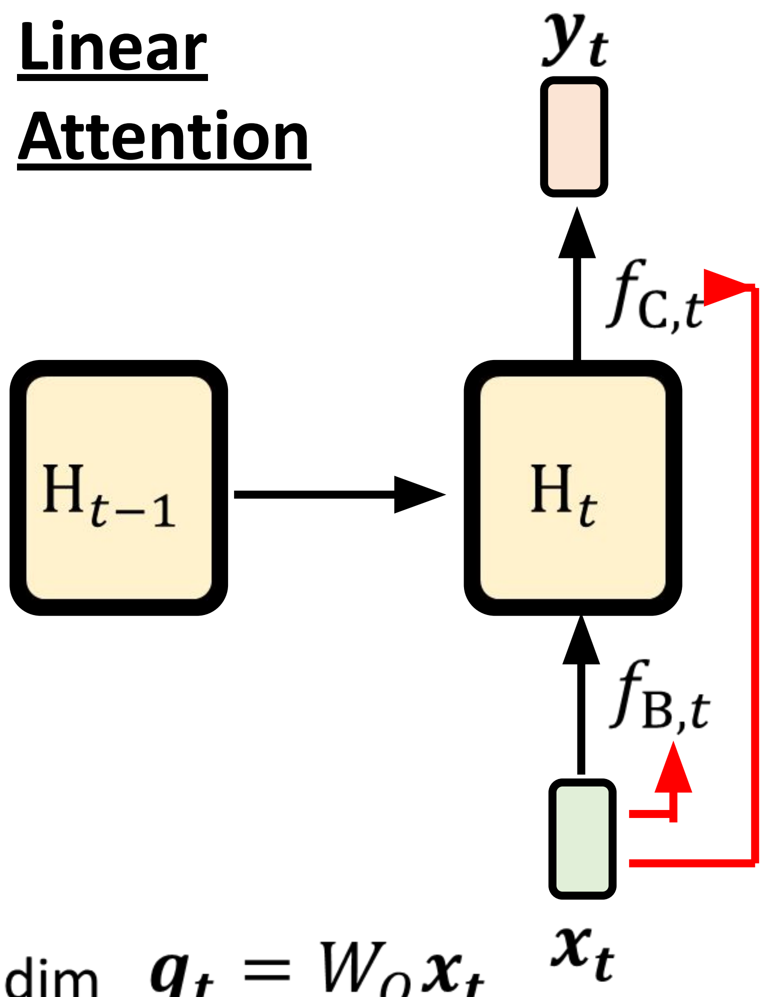
- Linear Attention 就是廣義 RNN 拿掉 "Reflection" $f_{A,t}$
- Linear Attention 就是 Self-attention 沒有 Softmax

Linear Attention

Training 的時候像 Self-attention
Inference 的時候像 RNN

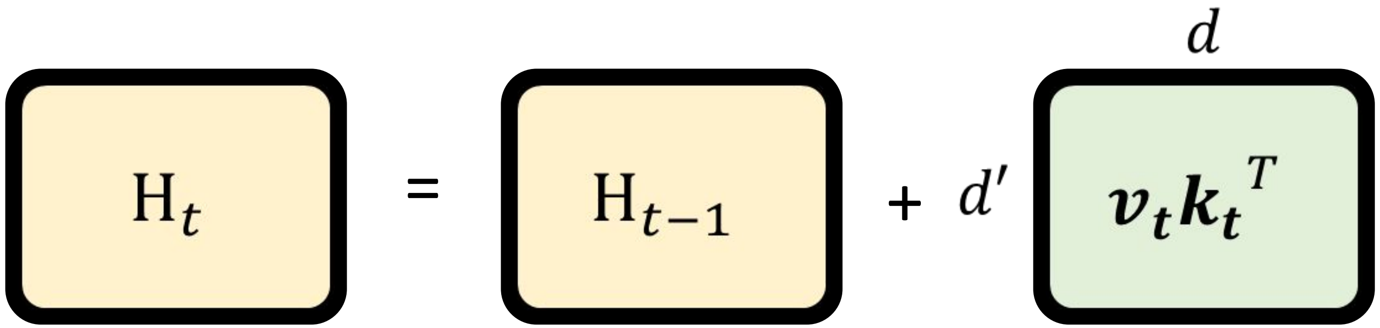


Linear Attention



$$H_t = H_{t-1} + f_{B,t}(\boldsymbol{x}_t) \qquad f_{B,t}(\boldsymbol{x}_t) = \boldsymbol{v}_t \boldsymbol{k}_t^T$$

$$\boldsymbol{y}_t = f_{C,t}(H_t) \qquad f_{C,t}(H_t) = H_t \boldsymbol{q}_t$$

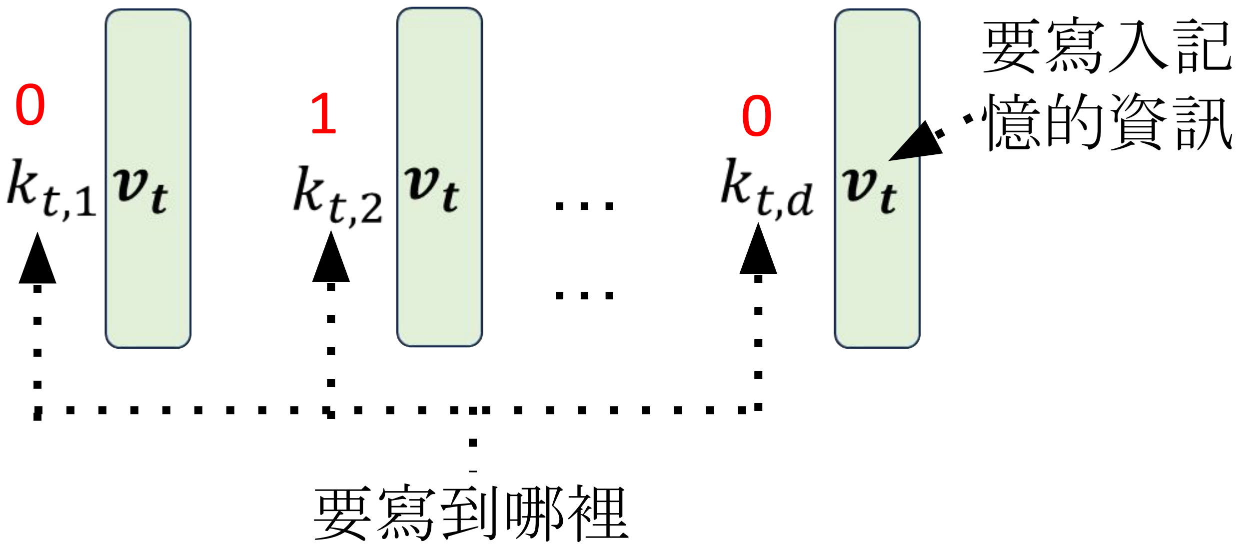


d dim $\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$

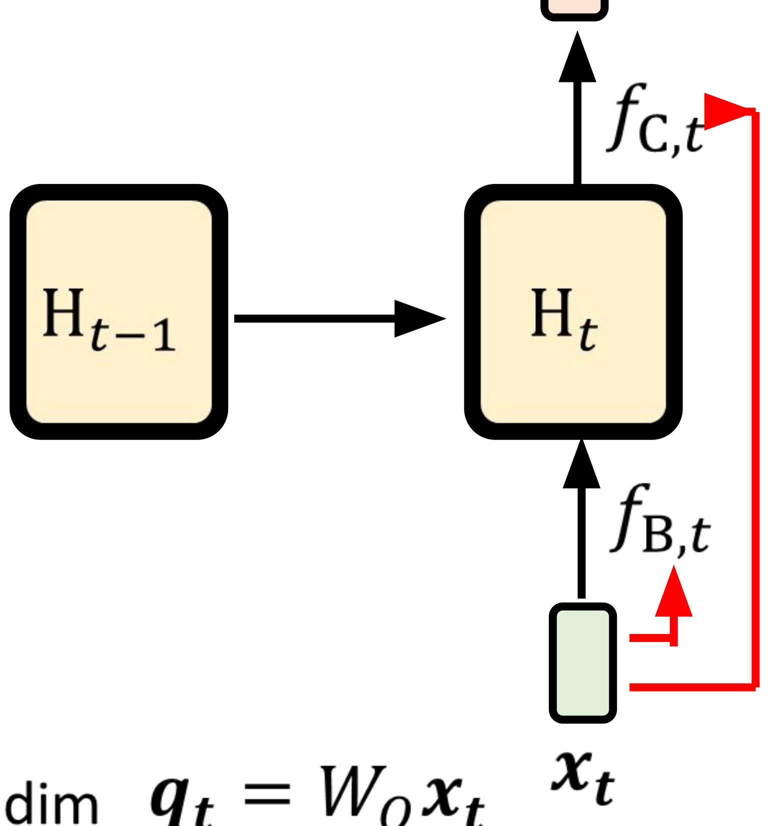
d dim $\boldsymbol{k}_t = W_k \boldsymbol{x}_t$

d' dim $\boldsymbol{v}_t = W_v \boldsymbol{x}_t$

把 $\boldsymbol{v}_t$ 寫入 H 的 2nd column

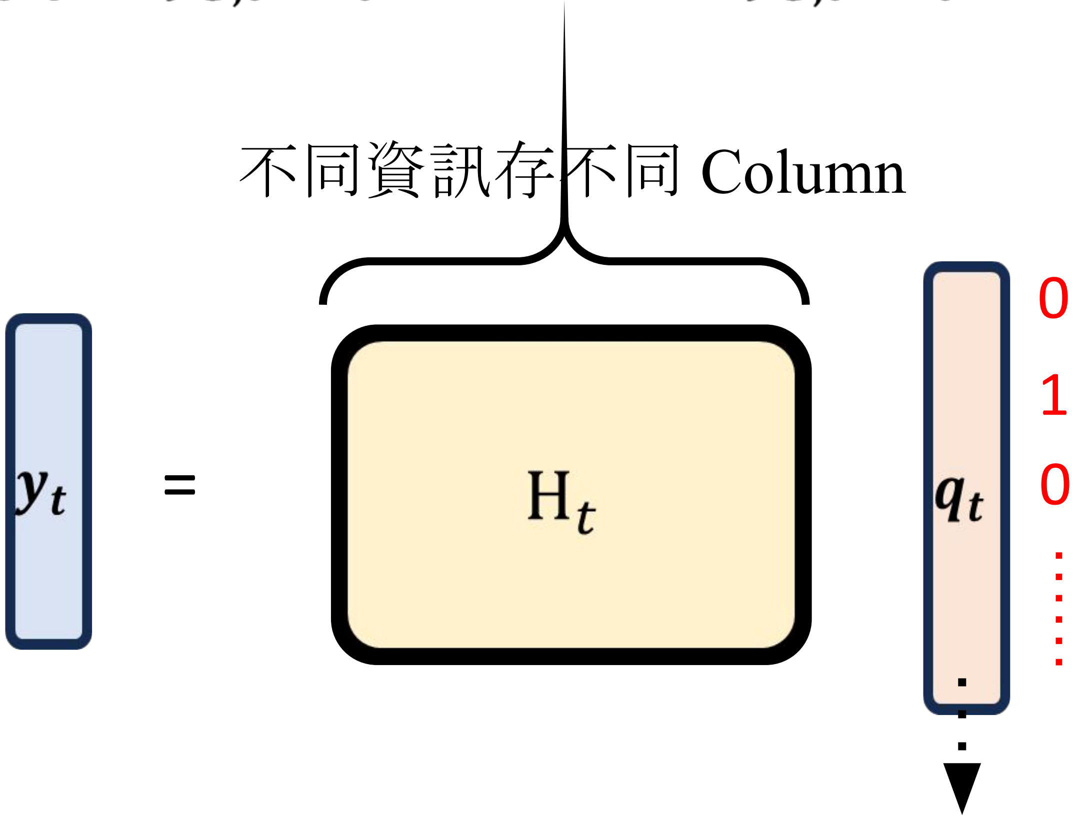


Linear Attention



$$H_t = H_{t-1} + f_{B,t}(\boldsymbol{x}_t) \qquad f_{B,t}(\boldsymbol{x}_t) = \boldsymbol{v}_t \boldsymbol{k}_t^T$$

$$\boldsymbol{y}_t = f_{C,t}(H_t) \qquad f_{C,t}(H_t) = H_t \boldsymbol{q}_t$$



d dim $\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$

d dim $\boldsymbol{k}_t = W_k \boldsymbol{x}_t$

d' dim $\boldsymbol{v}_t = W_v \boldsymbol{x}_t$

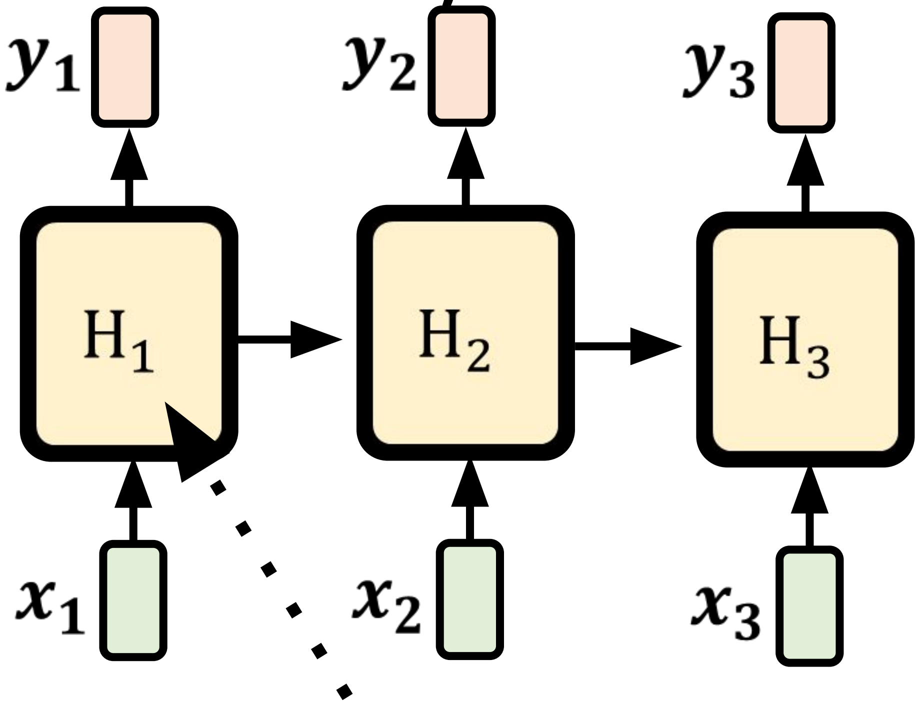
這不是甚麼新想法 ……

Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention
https://arxiv.org/abs/2006.16236

https://youtu.be/yHoAq1IT_og?si=pSymySFnZqQj51Ik

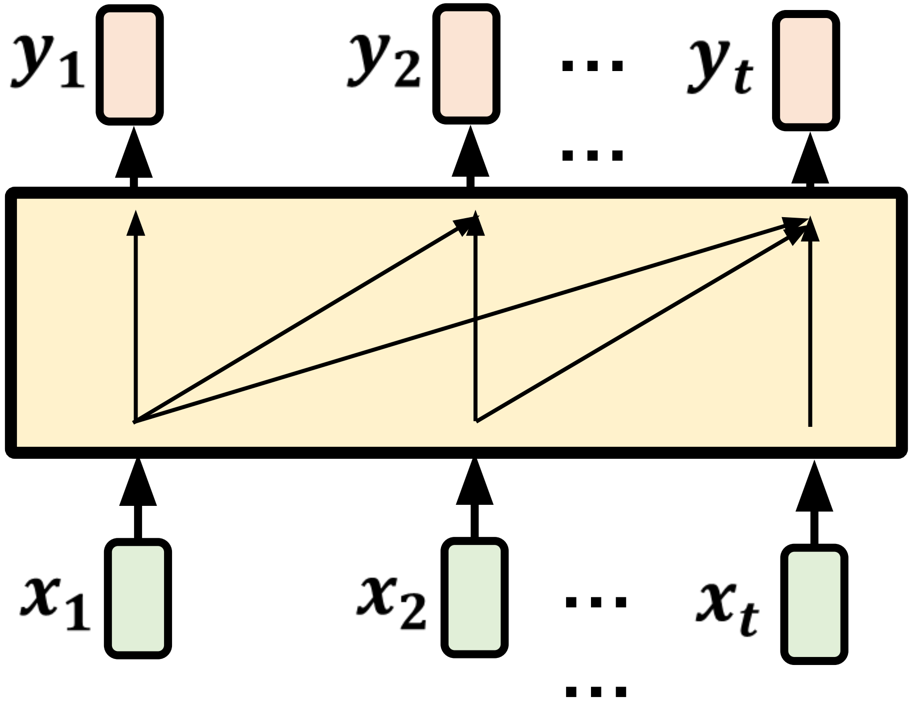
RNN (Linear Attention) 贏不過 Transformer (Self-attention with Softmax)？

RNN (Linear Attention)



~~記憶太小~~ 記憶有限

Transformer (Self-attention with softmax)



無限記憶?

RNN (Linear Attention) 贏不過 Transformer (Self-attention with Softmax)？

RNN (Linear Attention)

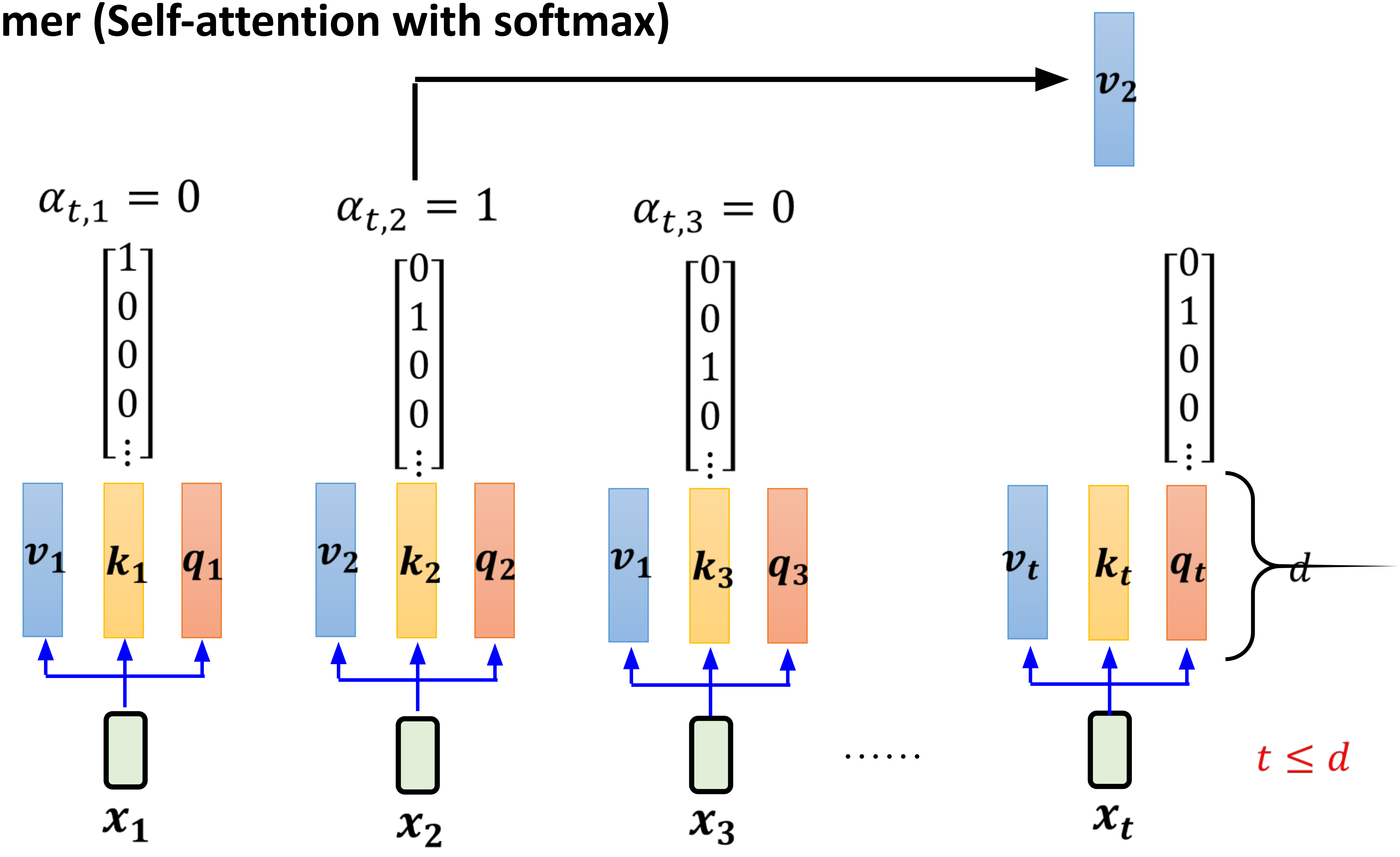
$$\mathrm{H}_t = \mathrm{H}_{t-1} + \boldsymbol{v}_t {\boldsymbol{k}_t}^T \qquad \boldsymbol{y}_t = \mathrm{H}_t \boldsymbol{q}_t$$

d

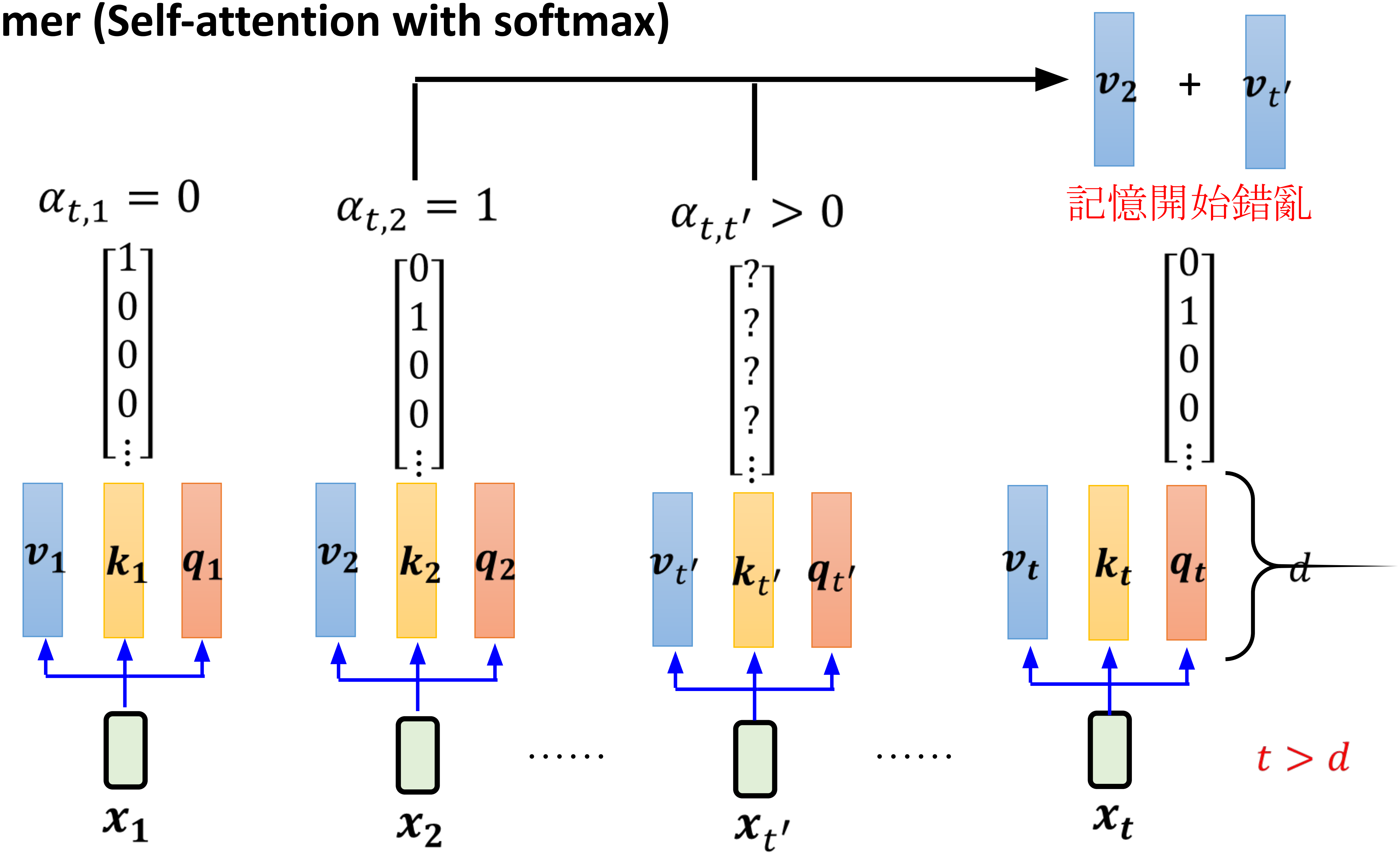
$\boldsymbol{v}_1$ $\boldsymbol{v}_2$ $\boldsymbol{v}_3$ … 最多存 d 個 $\boldsymbol{v}$ 不受干擾

$${\boldsymbol{k}_1}^T = [1 \quad 0 \quad ...] \qquad {\boldsymbol{k}_2}^T = [0 \quad 1 \quad ...] \qquad {\boldsymbol{k}_3}^T = [0 \quad 0 \quad 1 ...]$$

Transformer (Self-attention with softmax)



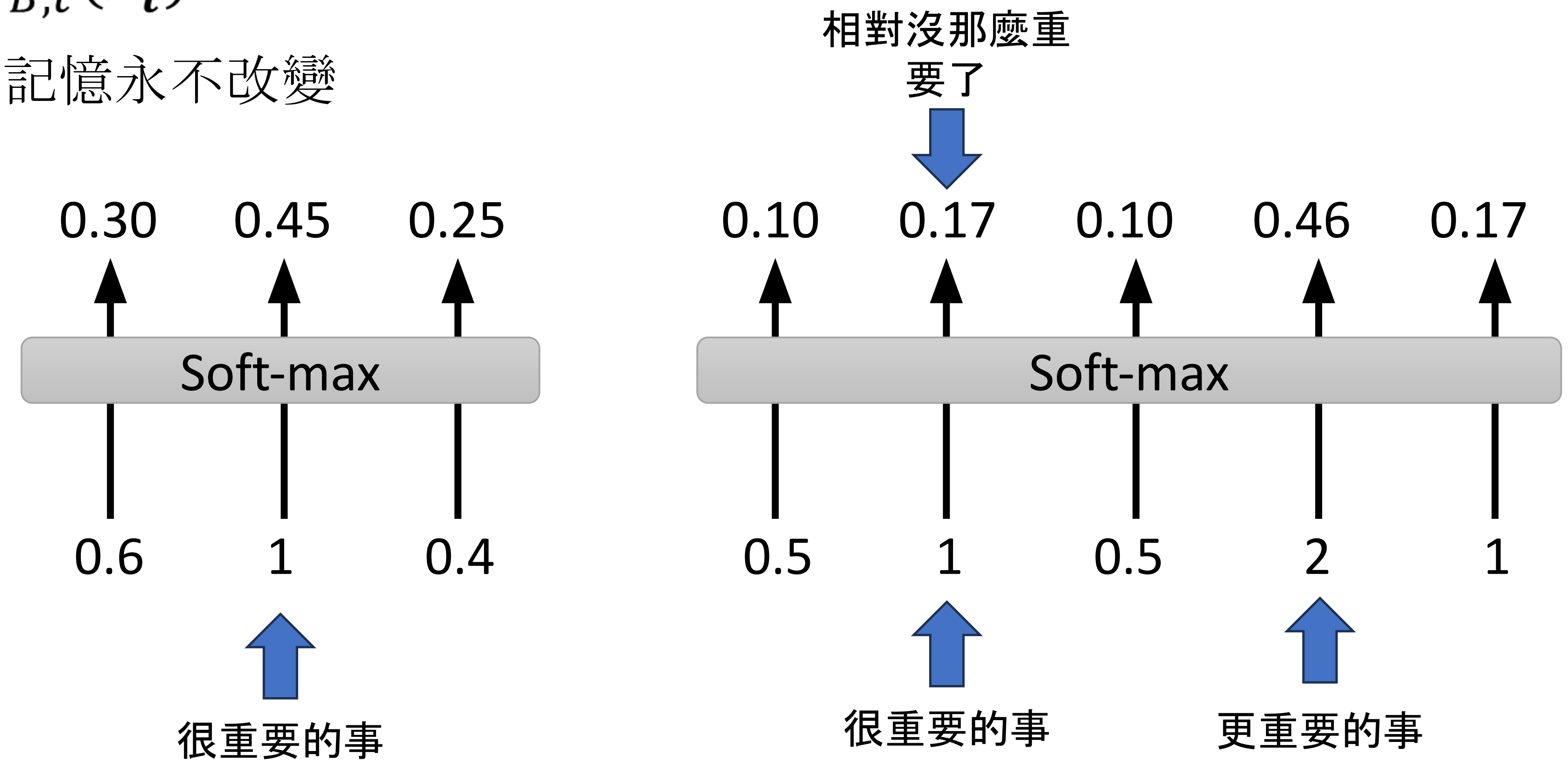
Transformer (Self-attention with softmax)



RNN (Linear Attention) 贏不過 Transformer (Self-attention with Softmax)？

$$\mathrm{H}_t = \mathrm{H}_{t-1} + f_{B,t}(\boldsymbol{x}_t)$$

Linear Attention 記憶永不改變



加上 Reflection: 逐渐遗忘

Linear Attention

$$\mathrm{H}_t = \mathrm{H}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t{}^T$$

$$\boldsymbol{y}_t = \mathrm{H}_t \boldsymbol{q}_t$$

$$\boldsymbol{v}_t = W_v \boldsymbol{x}_t$$

$$\boldsymbol{k}_t = W_k \boldsymbol{x}_t$$

$$\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$$

Retention Network (RetNet)

$$\mathrm{H}_t = \gamma \mathrm{H}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t{}^T$$

$$\boldsymbol{y}_t = \mathrm{H}_t \boldsymbol{q}_t$$

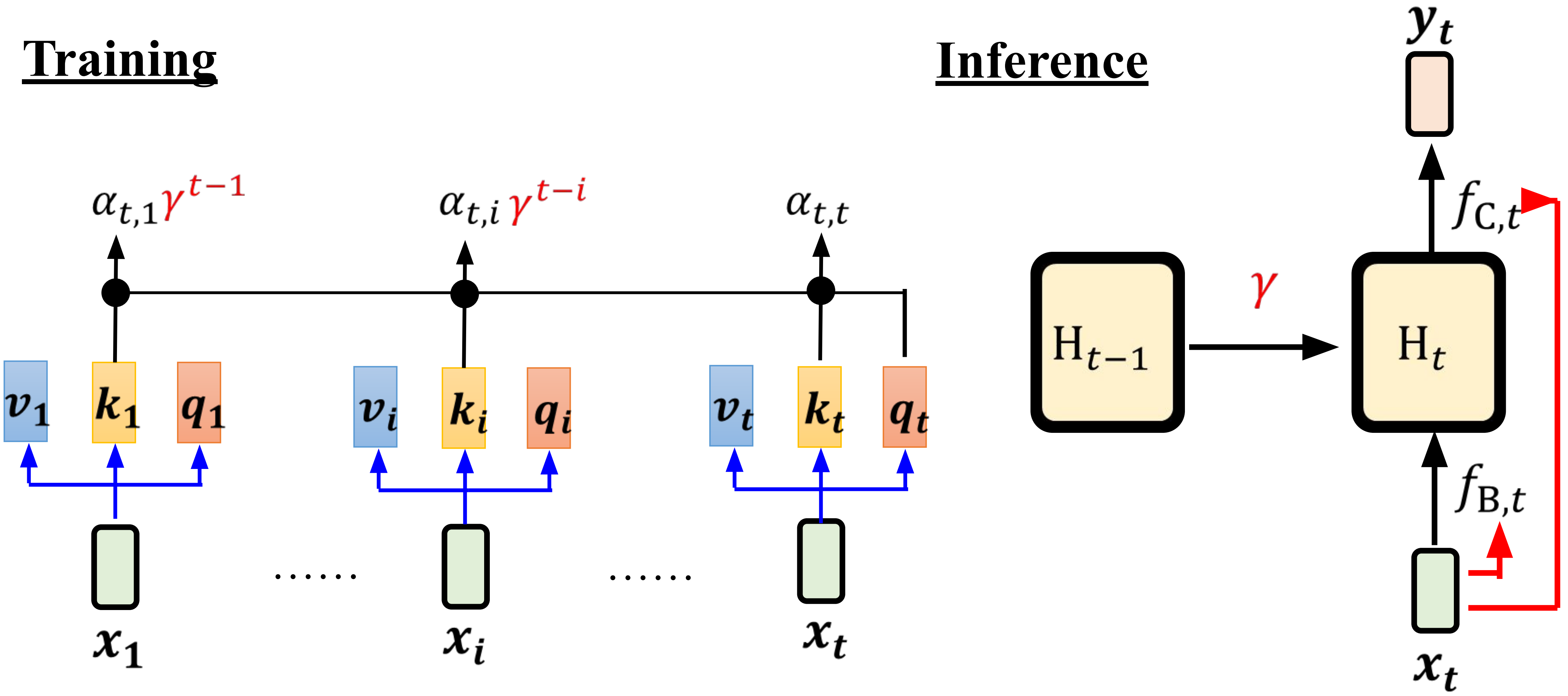
$$\boldsymbol{v}_t = W_v \boldsymbol{x}_t$$

$$\boldsymbol{k}_t = W_k \boldsymbol{x}_t$$

$$\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$$

https://arxiv.org/abs/2307.08621

加上 Reflection: 逐漸遺忘



加上 Reflection: 根據情況遺忘

https://arxiv.org/abs/2405.05254

Retention Network (RetNet)

$$\mathrm{H}_t = \gamma \mathrm{H}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t{}^T$$

$$\boldsymbol{y}_t = \mathrm{H}_t \boldsymbol{q}_t$$

$$\boldsymbol{v}_t = W_v \boldsymbol{x}_t$$

$$\boldsymbol{k}_t = W_k \boldsymbol{x}_t$$

$$\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$$

Gated Retention

$$\mathrm{H}_t = \gamma_t \mathrm{H}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t{}^T$$

$$\boldsymbol{y}_t = \mathrm{H}_t \boldsymbol{q}_t$$

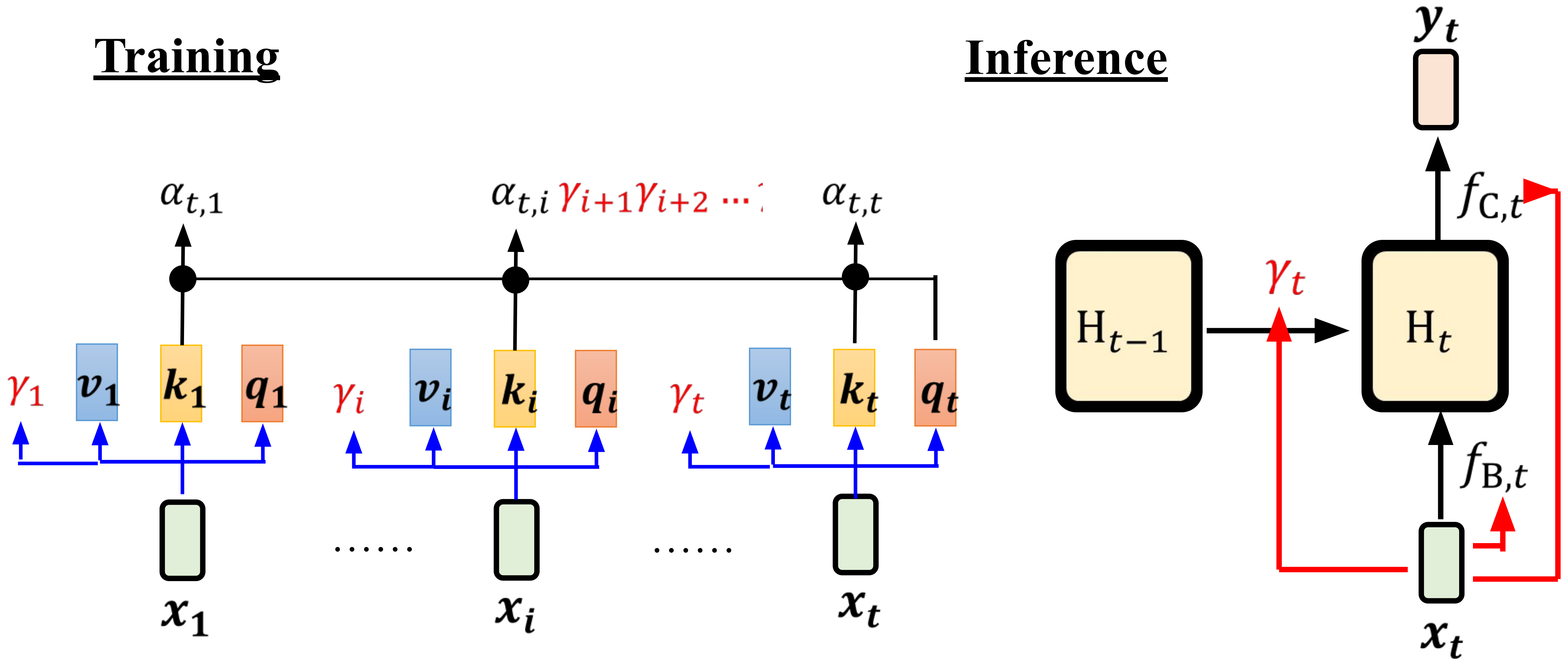
$$\boldsymbol{v}_t = W_v \boldsymbol{x}_t$$

$$\boldsymbol{k}_t = W_k \boldsymbol{x}_t$$

$$\boldsymbol{q}_t = W_Q \boldsymbol{x}_t$$

$$\gamma_t = sigmoid(W_\gamma \boldsymbol{x}_t)$$

加上 Reflection: 逐漸遺忘



更複雜的 Reflection

$$\boldsymbol{s_t}^T = [0 \quad 1 \quad 0.1 \quad \ldots \ldots]$$

$$\mathrm{H}_t = \color{blue}{G_t} \color{black} \odot \mathrm{H}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t{}^T$$

$$\color{blue}{G_t} \color{black} = \boldsymbol{e}_t \boldsymbol{s}_t{}^T$$

$$\color{blue}{G_t} \color{black} = \boldsymbol{1} \boldsymbol{s}_t{}^T$$

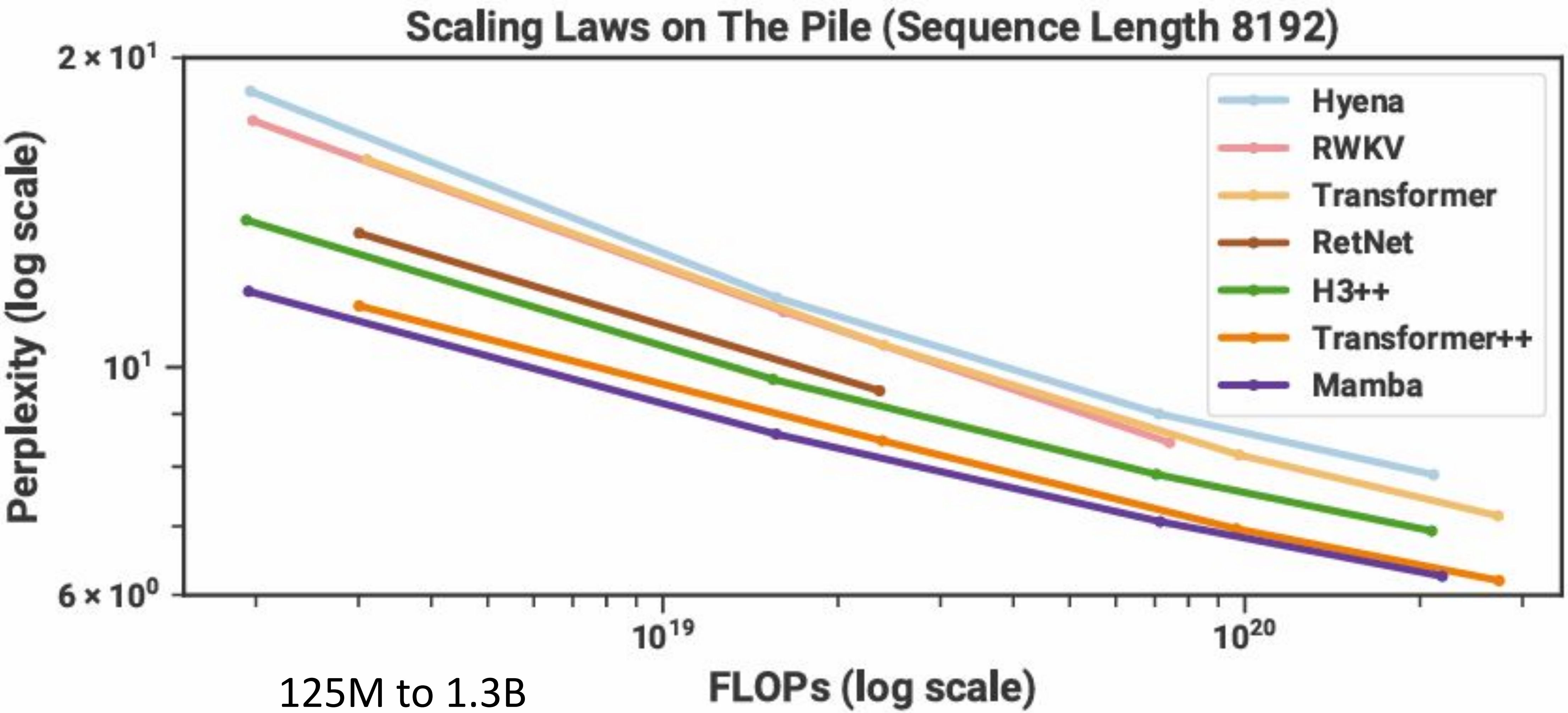
$$\boldsymbol{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$\begin{array}{cc} & \boldsymbol{s_t}^T \\ \boldsymbol{1} & \begin{bmatrix} s_{t,1} & s_{t,2} & s_{t,3} & & s_{t,d} \\ s_{t,1} & s_{t,2} & s_{t,3} & & s_{t,d} \\ \vdots & \vdots \quad \vdots & \vdots & \cdots \vdots & \vdots \\ s_{t,1} & s_{t,2} & s_{t,3} & & s_{t,d} \end{bmatrix} \odot \mathrm{H}_{t-1} \\ & \begin{array}{ccc} 0 & 1 & 0.1 \end{array} \end{array}$$

抹去 保留 減弱

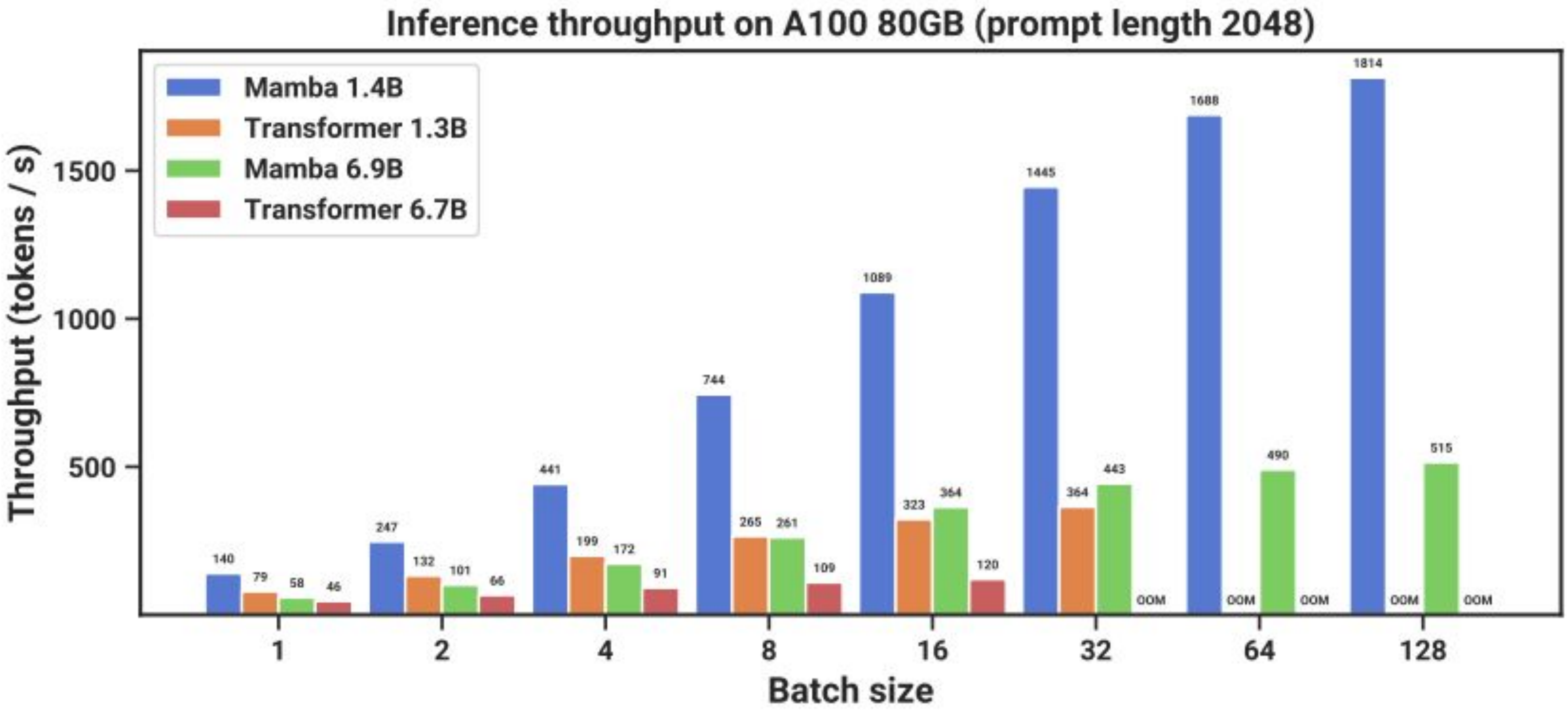
Model	Recurrence	Memory read-out
Linear Attention [48, 47]	$\mathbf{S}_t = \mathbf{S}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
+ Kernel	$\mathbf{S}_t = \mathbf{S}_{t-1} + \boldsymbol{v}_t \phi(\boldsymbol{k}_t)^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \phi(\boldsymbol{q}_t)$
+ Normalization	$\mathbf{S}_t = \mathbf{S}_{t-1} + \boldsymbol{v}_t \phi(\boldsymbol{k}_t)^\top, \quad \boldsymbol{z}_t = \boldsymbol{z}_{t-1} + \phi(\boldsymbol{k}_t)$	$\boldsymbol{o}_t = \mathbf{S}_t \phi(\boldsymbol{q}_t) / (\boldsymbol{z}_t^\top \phi(\boldsymbol{q}_t))$
DeltaNet [101]	$\mathbf{S}_t = \mathbf{S}_{t-1}(\mathbf{I} - \beta_t \boldsymbol{k}_t \boldsymbol{k}_t^\top) + \beta_t \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
Gated RFA [81]	$\mathbf{S}_t = g_t \mathbf{S}_{t-1} + (1 - g_t) \boldsymbol{v}_t \boldsymbol{k}_t^\top, \quad \boldsymbol{z}_t = g_t \boldsymbol{z}_{t-1} + (1 - g_t) \boldsymbol{k}_t$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t / (\boldsymbol{z}_t^\top \boldsymbol{q}_t)$
S4 [32, 106]	$\mathbf{S}_t = \mathbf{S}_{t-1} \odot \exp(-(\boldsymbol{\alpha} \mathbf{1}^\top) \odot \exp(\boldsymbol{A})) + \boldsymbol{B} \odot (\boldsymbol{v}_t \mathbf{1}^\top)$	$\boldsymbol{o}_t = (\mathbf{S}_t \odot \boldsymbol{C}) \mathbf{1} + \boldsymbol{d} \odot \boldsymbol{v}_t$
ABC [82]	$\mathbf{S}_t^{\boldsymbol{k}} = \mathbf{S}_{t-1}^{\boldsymbol{k}} + \boldsymbol{k}_t \boldsymbol{\phi}_t^\top, \quad \mathbf{S}_t^{\boldsymbol{v}} = \mathbf{S}_{t-1}^{\boldsymbol{v}} + \boldsymbol{v}_t \boldsymbol{\phi}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t^{\boldsymbol{v}} \operatorname{softmax}\left(\mathbf{S}_t^{\boldsymbol{k}} \boldsymbol{q}_t\right)$
DFW [63]	$\mathbf{S}_t = \mathbf{S}_{t-1} \odot (\boldsymbol{\beta}_t \boldsymbol{\alpha}_t^\top) + \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
RetNet [108]	$\mathbf{S}_t = \gamma \mathbf{S}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
Mamba [31]	$\mathbf{S}_t = \mathbf{S}_{t-1} \odot \exp(-(\boldsymbol{\alpha}_t \mathbf{1}^\top) \odot \exp(\boldsymbol{A})) + (\boldsymbol{\alpha}_t \odot \boldsymbol{v}_t) \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t + \boldsymbol{d} \odot \boldsymbol{v}_t$
GLA [124]	$\mathbf{S}_t = \mathbf{S}_{t-1} \odot (\mathbf{1} \boldsymbol{\alpha}_t^\top) + \boldsymbol{v}_t \boldsymbol{k}_t^\top = \mathbf{S}_{t-1} \operatorname{Diag}(\boldsymbol{\alpha}_t) + \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
RWKV-6 [79]	$\mathbf{S}_t = \mathbf{S}_{t-1} \operatorname{Diag}(\boldsymbol{\alpha}_t) + \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = (\mathbf{S}_{t-1} + (\boldsymbol{d} \odot \boldsymbol{v}_t) \boldsymbol{k}_t^\top) \boldsymbol{q}_t$
HGRN-2 [92]	$\mathbf{S}_t = \mathbf{S}_{t-1} \operatorname{Diag}(\boldsymbol{\alpha}_t) + \boldsymbol{v}_t (\mathbf{1} - \boldsymbol{\alpha}_t)^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
mLSTM [9]	$\mathbf{S}_t = f_t \mathbf{S}_{t-1} + i_t \boldsymbol{v}_t \boldsymbol{k}_t^\top, \quad \boldsymbol{z}_t = f_t \boldsymbol{z}_{t-1} + i_t \boldsymbol{k}_t$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t / \max\{1, \lvert \boldsymbol{z}_t^\top \boldsymbol{q}_t \rvert\}$
Mamba-2 [19]	$\mathbf{S}_t = \gamma_t \mathbf{S}_{t-1} + \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$
GSA [131]	$\mathbf{S}_t^{\boldsymbol{k}} = \mathbf{S}_{t-1}^{\boldsymbol{k}} \operatorname{Diag}(\boldsymbol{\alpha}_t) + \boldsymbol{k}_t \boldsymbol{\phi}_t^\top, \quad \mathbf{S}_t^{\boldsymbol{v}} = \mathbf{S}_{t-1}^{\boldsymbol{v}} \operatorname{Diag}(\boldsymbol{\alpha}_t) + \boldsymbol{v}_t \boldsymbol{\phi}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t^{\boldsymbol{v}} \operatorname{softmax}\left(\mathbf{S}_t^{\boldsymbol{k}} \boldsymbol{q}_t\right)$
Gated DeltaNet [125]	$\mathbf{S}_t = \mathbf{S}_{t-1} \left(\alpha_t (\mathbf{I} - \beta_t \boldsymbol{k}_t \boldsymbol{k}_t^\top)\right) + \beta_t \boldsymbol{v}_t \boldsymbol{k}_t^\top$	$\boldsymbol{o}_t = \mathbf{S}_t \boldsymbol{q}_t$

https://arxiv.org/abs/2406.06484



125M to 1.3B

https://arxiv.org/abs/2312.00752



https://arxiv.org/abs/2312.00752

DeltaNet

https://arxiv.org/abs/2406.06484

$$\mathrm{H}_t = \mathrm{H}_{t-1}\left(I - \beta_t \boldsymbol{k_t} \boldsymbol{k_t}^T\right) + \beta_t \boldsymbol{v_t} \boldsymbol{k_t}^T$$

$$\mathrm{H}_t = \mathrm{H}_{t-1} + \boldsymbol{v_t} \boldsymbol{k_t}^T$$

也是 Gradient Descent，只是 L_t 不一樣

$$\mathrm{H}_t = \mathrm{H}_{t-1} - \boldsymbol{v_{t,old}} \boldsymbol{k_t}^T + \boldsymbol{v_t} \boldsymbol{k_t}^T \qquad \boldsymbol{v_{t,old}} = \mathrm{H}_{t-1} \boldsymbol{k_t}$$

$$\mathrm{H}_t = \mathrm{H}_{t-1} - \beta_t \boldsymbol{v_{t,old}} \boldsymbol{k_t}^T + \beta_t \boldsymbol{v_t} \boldsymbol{k_t}^T$$

$$\mathrm{H}_t = \mathrm{H}_{t-1} - \beta_t \mathrm{H}_{t-1} \boldsymbol{k_t} \boldsymbol{k_t}^T + \beta_t \boldsymbol{v_t} \boldsymbol{k_t}^T$$

Gradient Descent

$$\mathrm{H}_t = \mathrm{H}_{t-1} - \beta_t \underline{(\mathrm{H}_{t-1} \boldsymbol{k_t} - \boldsymbol{v_t}) \boldsymbol{k_t}^T}$$

Parameter after update | before update | learning rate | gradient

$$L_t(H) = \frac{1}{2} \|H \boldsymbol{k_t} - \boldsymbol{v_t}\|^2$$

$$\nabla L_t(\mathrm{H}_{t-1})$$

更新 H 使得用 $\boldsymbol{k_t}$ 抽取出的資訊和 $\boldsymbol{v_t}$ 越接近越好

Titans: Learning to Memorize at Test Time https://arxiv.org/abs/2501.00663

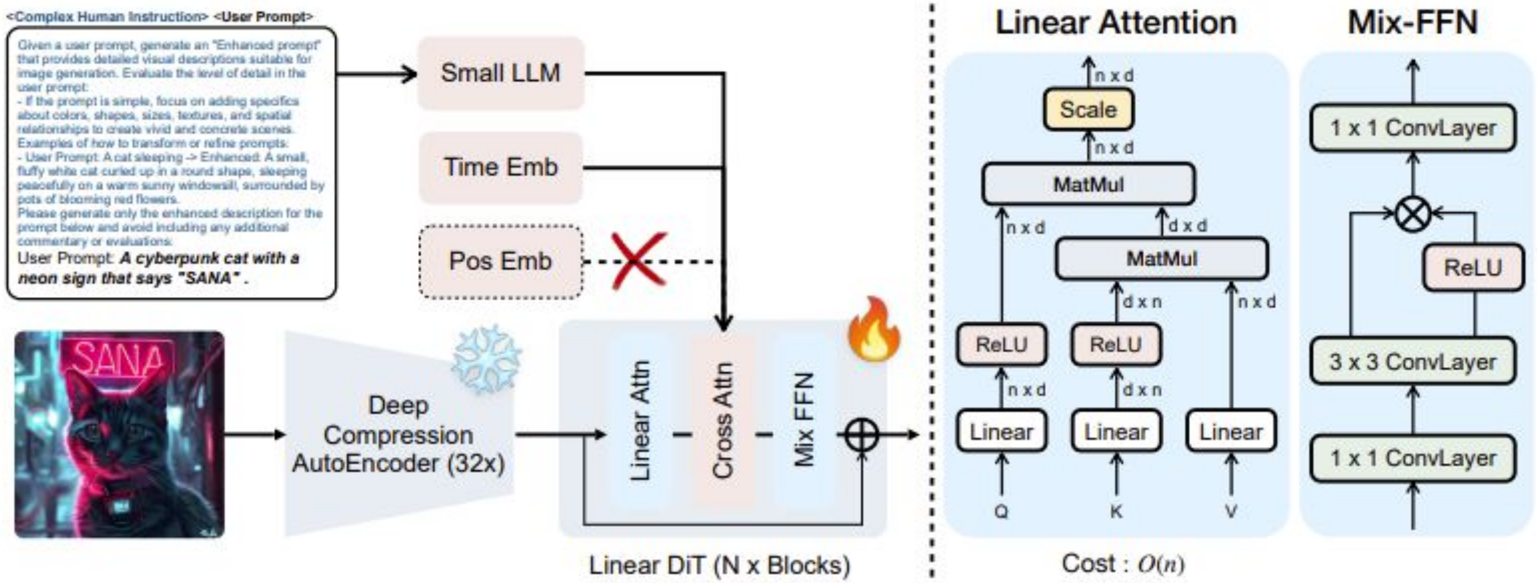
Name	Modality	Affiliations	Sizes	Access Link
Mamba 1&2	Language	Carnegie Mellon University & Princeton University	130M-2.8B	1
Falcon Mamba 7B	Language	Technology Innovation Institute	7B	2
Mistral 7B	Language	Mistral AI & NVIDIA	7B	3
Jamba	Language	AI21 Lab	12B/52B	4
Vision Mamba	Vision	Huazhong University of Science and Technology	7M-98M	5
VideoMamba	Video	OpenGVLab, Shanghai AI Laboratory	28M-392M	6
Codestral Mamba	Code	Mistral AI	7B, 22B	7

1. https://github.com/state-spaces/mamba
2. https://huggingface.co/tiiuae/falcon-mamba-7b
3. https://huggingface.co/mistralai/Mistral-7B-v0.1
4. https://huggingface.co/ai21labs/Jamba-v0.1
5. https://huggingface.co/hustvl/Vim-base-midclstok
6. https://huggingface.co/OpenGVLab/VideoMamba
7. https://mistral.ai/news/codestral-mamba/

https://arxiv.org/abs/2408.01129

Minimax-01

https://arxiv.org/abs/2501.08313



(a). Architecture overview of our Sana. (b). Linear DiT Module.

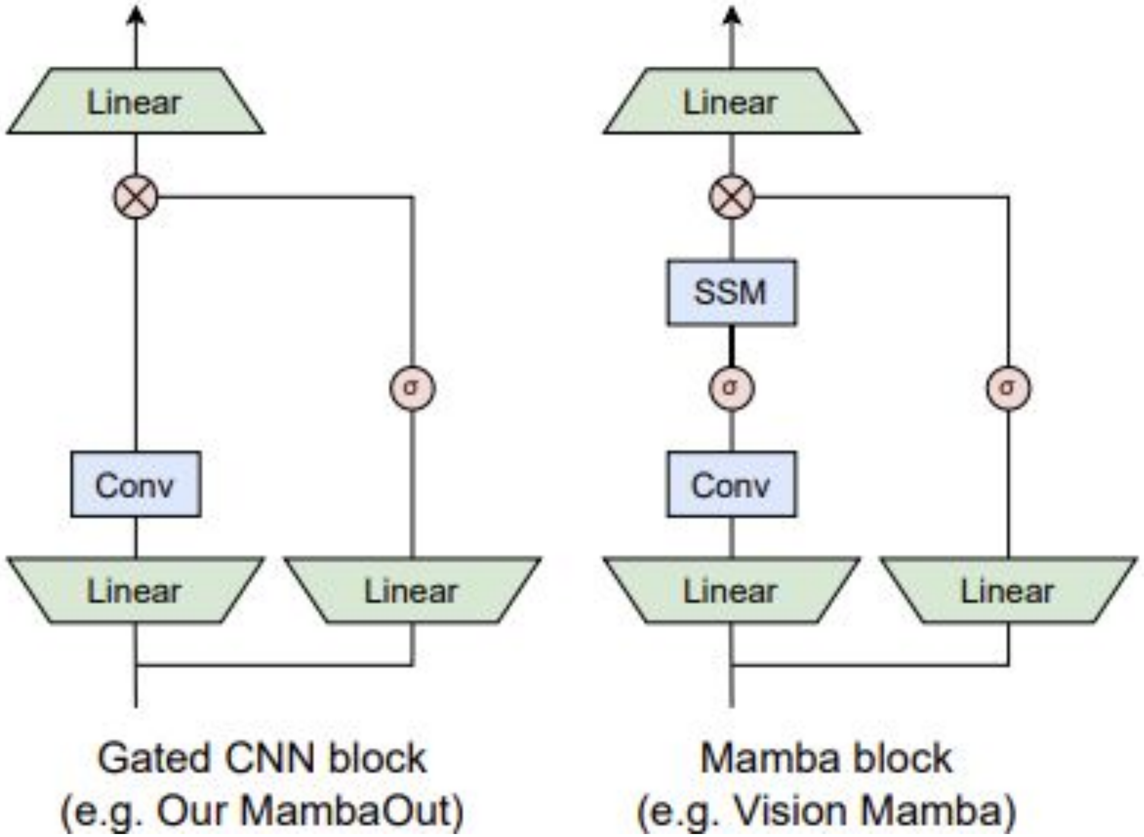
https://arxiv.org/abs/2410.10629

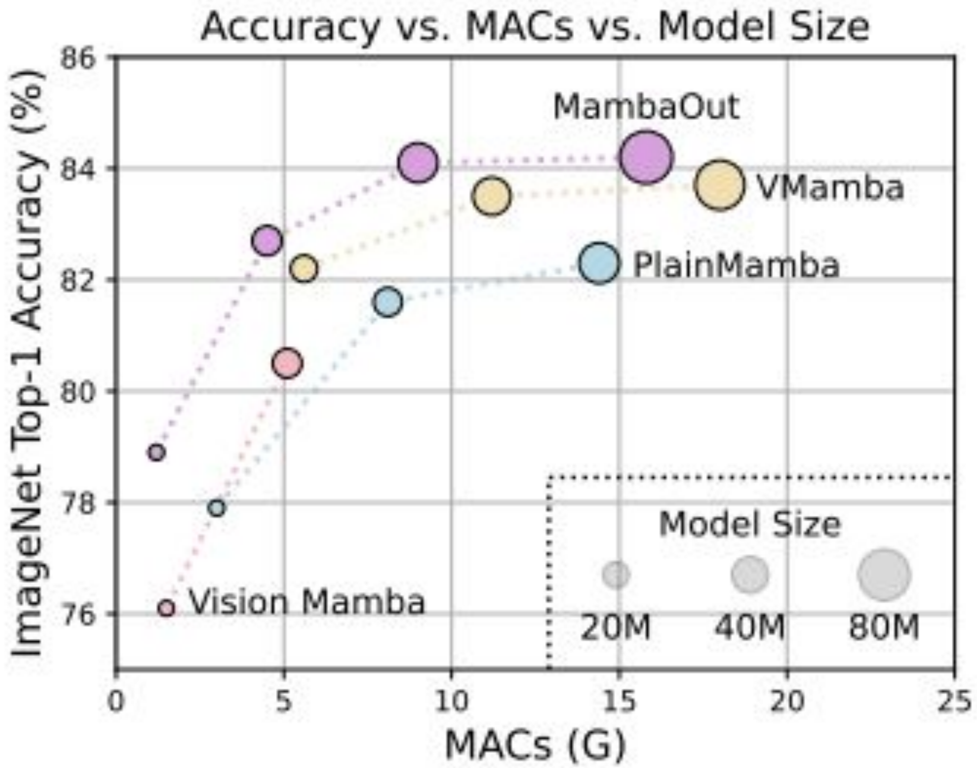
MambaOut: Do We Really Need Mamba for Vision?

https://arxiv.org/abs/2405.07992

In memory of Kobe Bryant

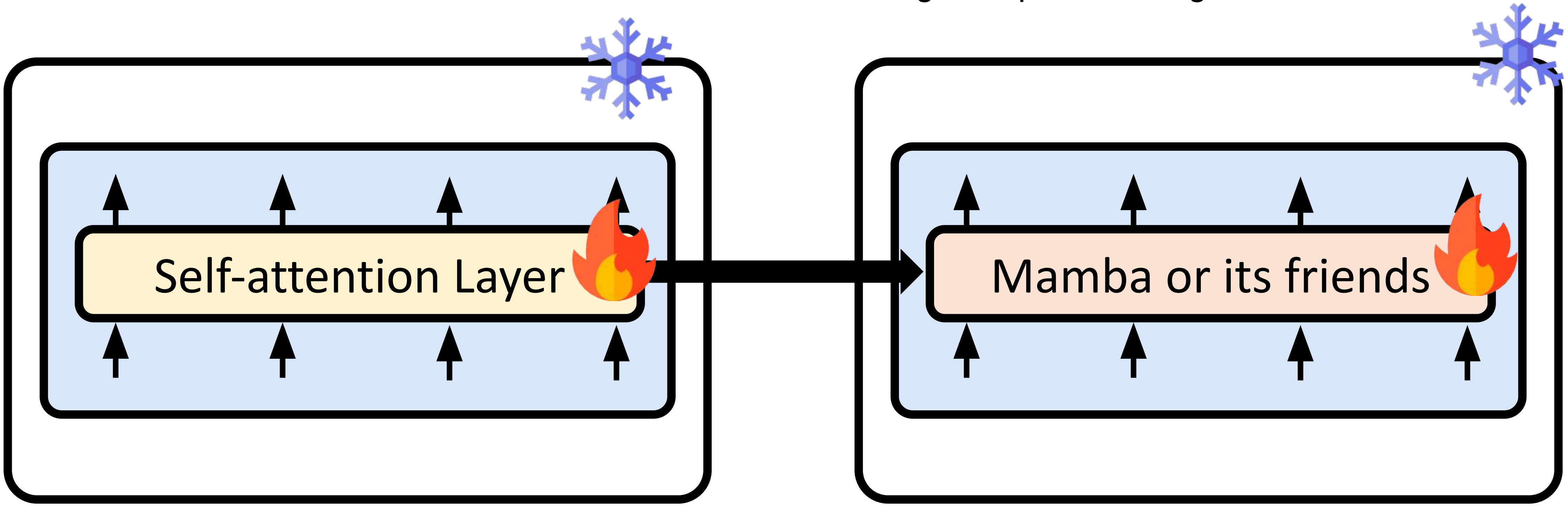
"What can I say, Mamba out." — *Kobe Bryant's NBA farewell speech in 2016.*





Do not train from scratch

Low-rank Linear Conversion via Attention Transfer (LoLCATs), https://arxiv.org/abs/2410.10254
The Mamba in the Llama, https://arxiv.org/abs/2408.15237
Transformers to SSMs, https://arxiv.org/abs/2408.10189
Linger, https://arxiv.org/abs/2503.01496



Is Attention All You Need?

Current Status: Yes

Time Remaining: 656d 19h 39m 37s

https://www.isattentionallyouneed.com/

Proposition:

On January 1, 2027, a Transformer-like model will continue to hold the state-of-the-art position in most benchmarked tasks in natural language processing.

For the Motion

Jonathan Frankle
@jefrankle
Harvard Professor
Chief Scientist Mosaic ML

Against the Motion

Sasha Rush
@srush_nlp
Cornell Professor
Research Scientist Hugging Face 🤗

https://www.isattentionallyouneed.com/