

Neural Networks III

Instructor: Siting Liu

School of Information Science and Technology (SIST), ShanghaiTech University

Shanghai Institute of Semiconductor Innovation (SISI)

<https://faculty.sist.shanghaitech.edu.cn/liust/EE219/EE219.html>



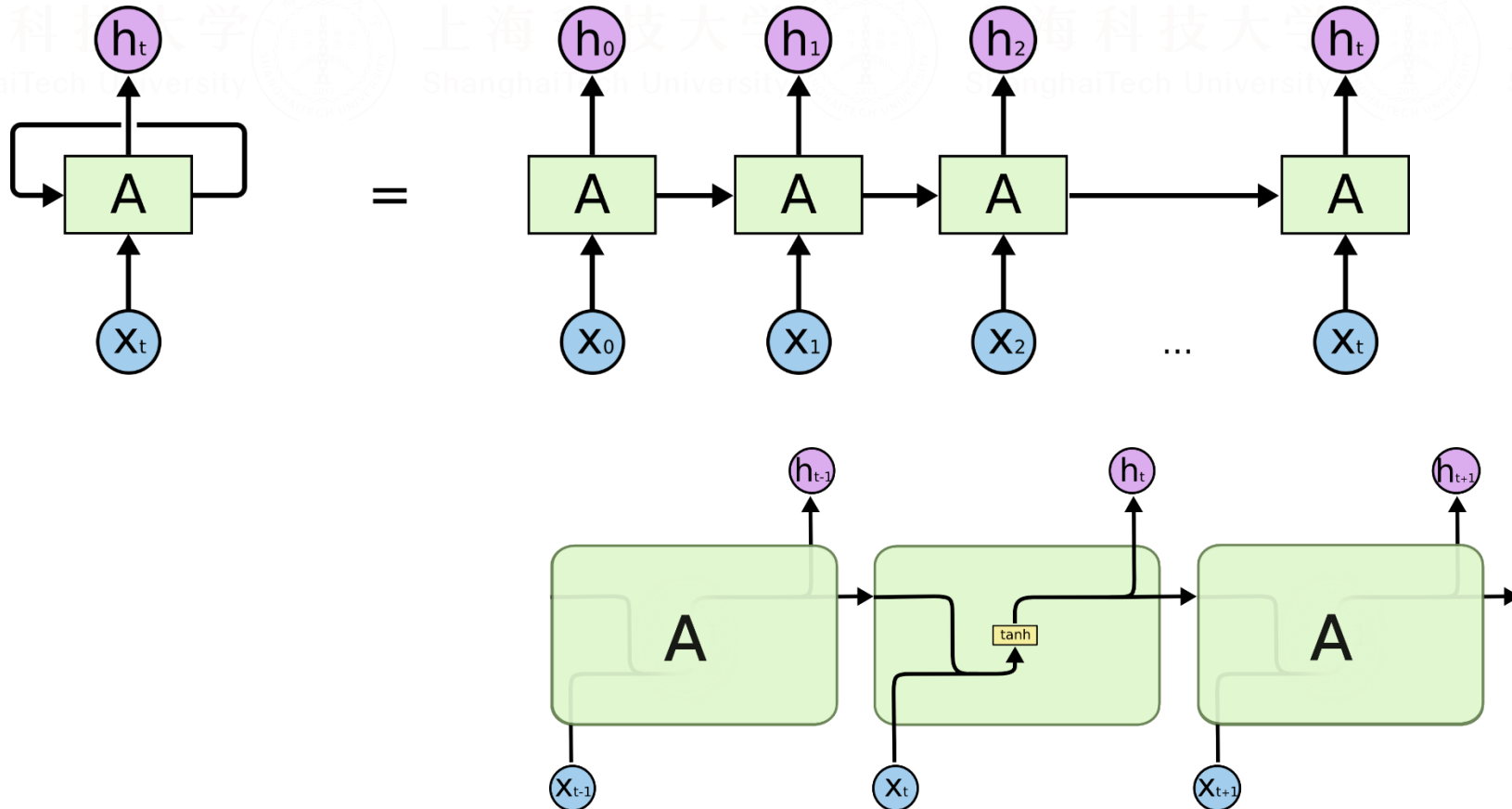
Computer Architecture - EE219
faculty.sist.shanghaitech.edu.cn

立志成才 报国裕民



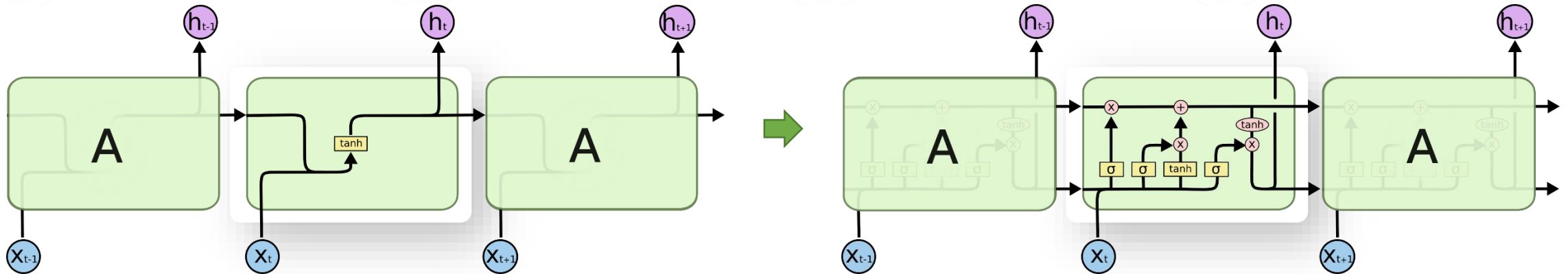
Feedback Connections - RNN

- Recurrent neural networks



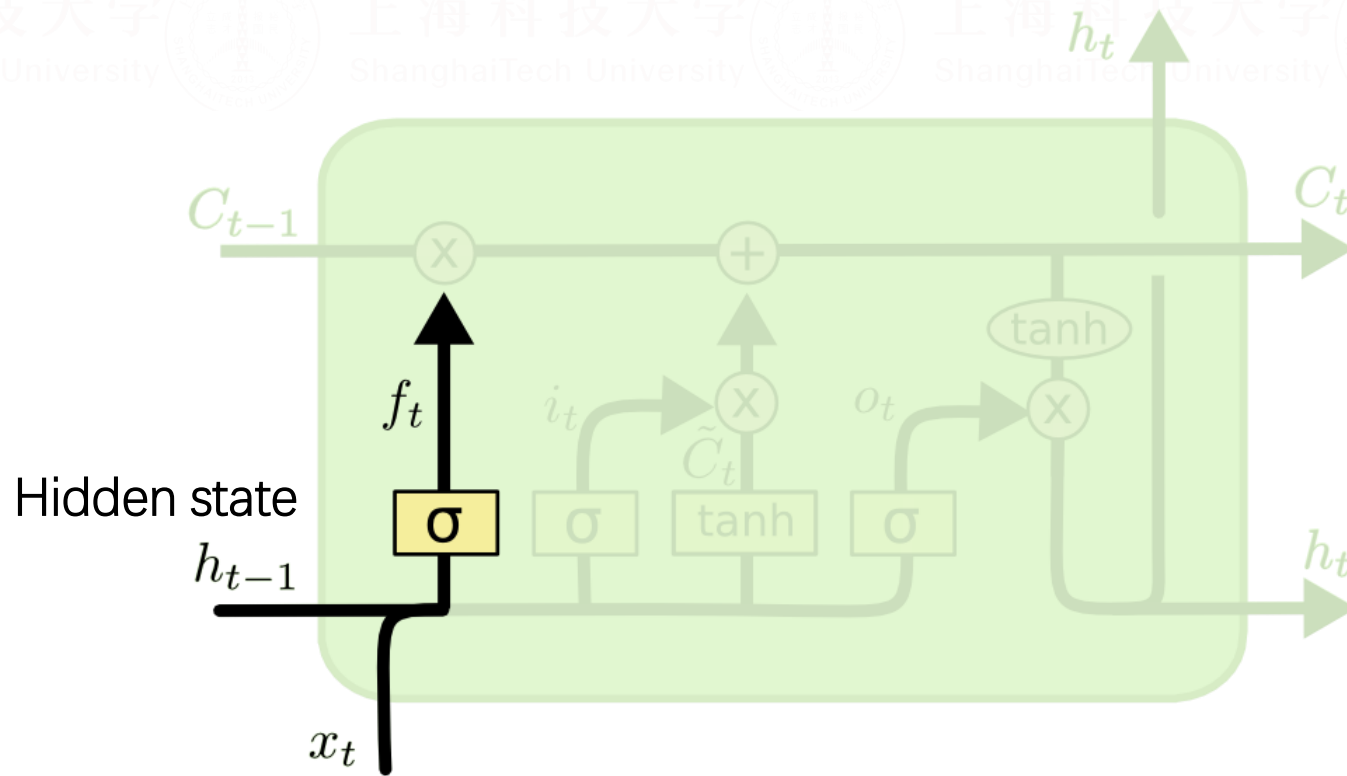
Feedback Connections - LSTM

- Long-short term memory (LSTM), stem from RNN



Gating Mechanism – Forget-gate

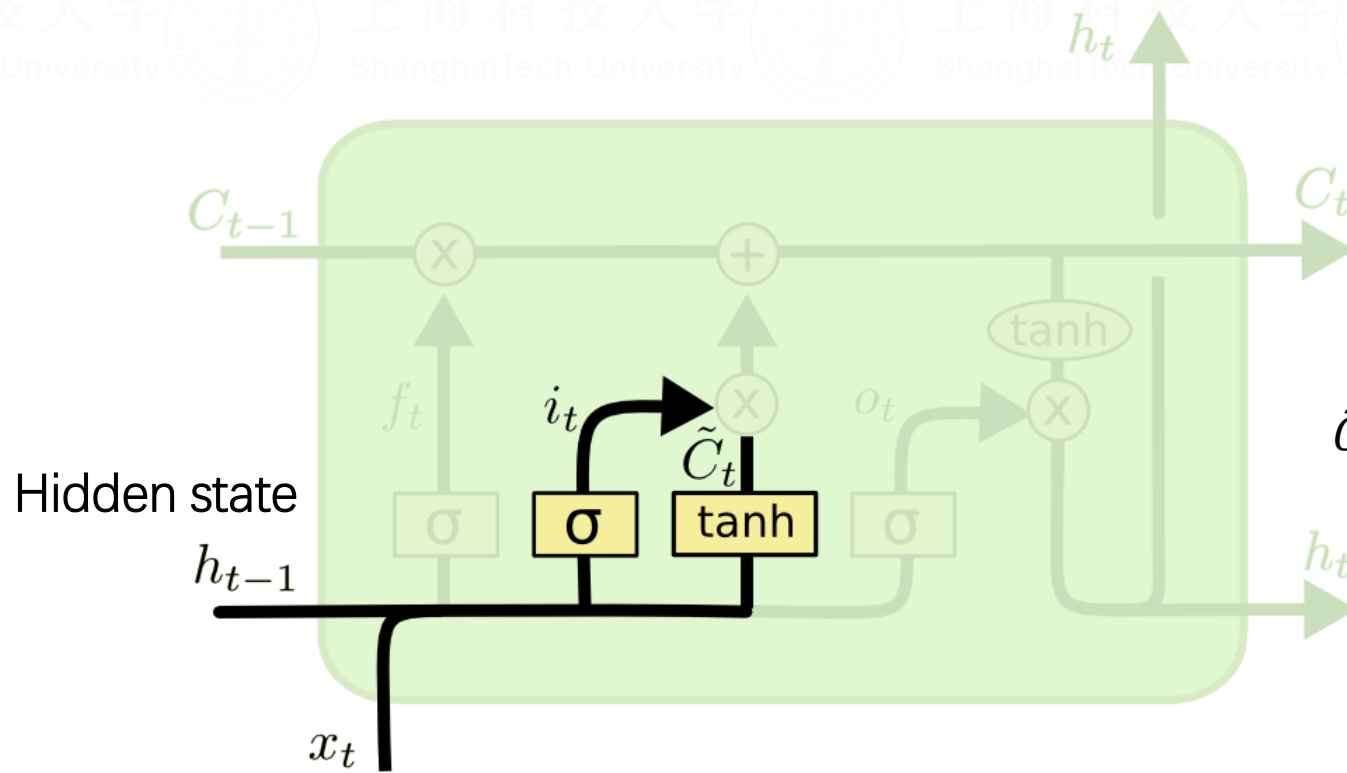
- A learnable parameter that dynamically decide how many information get through.



$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Gating Mechanism – Input-gate

- A learnable parameter that dynamically decide how many information get through.

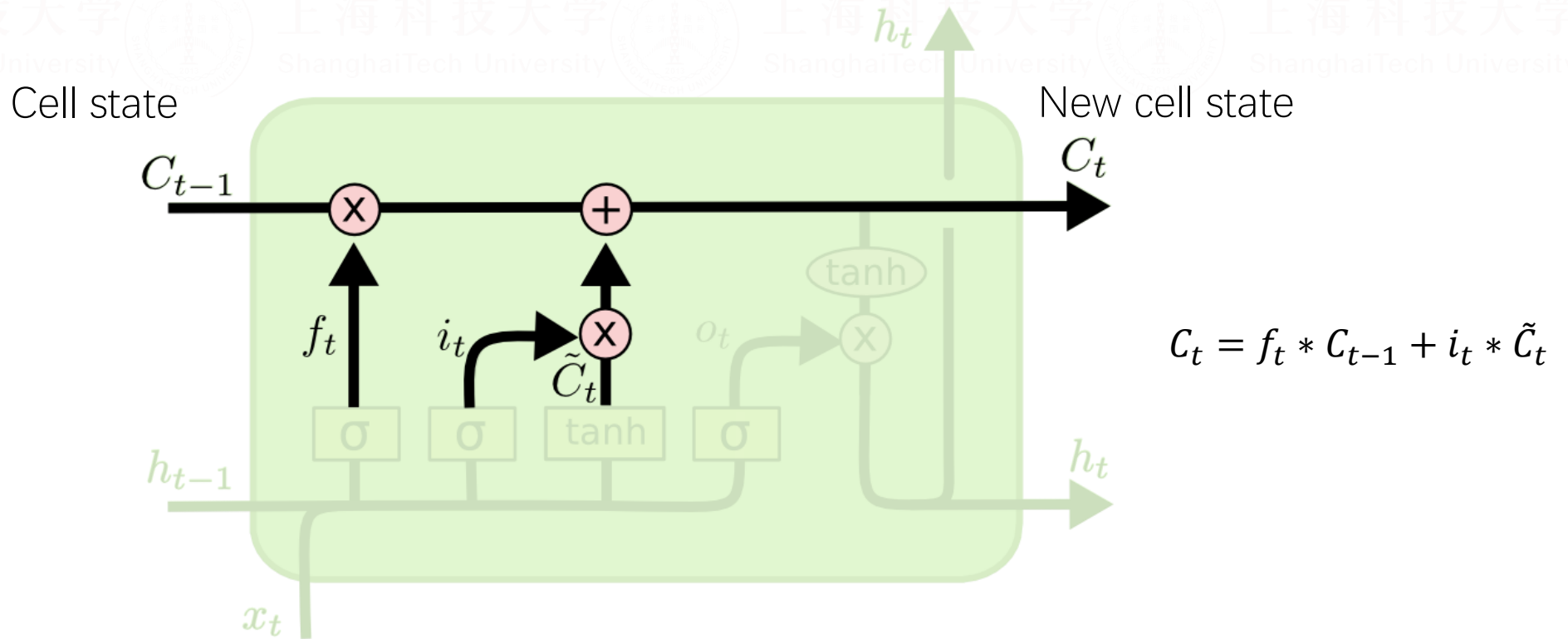


$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_i \cdot [h_{t-1}, x_t] + b_c)$$

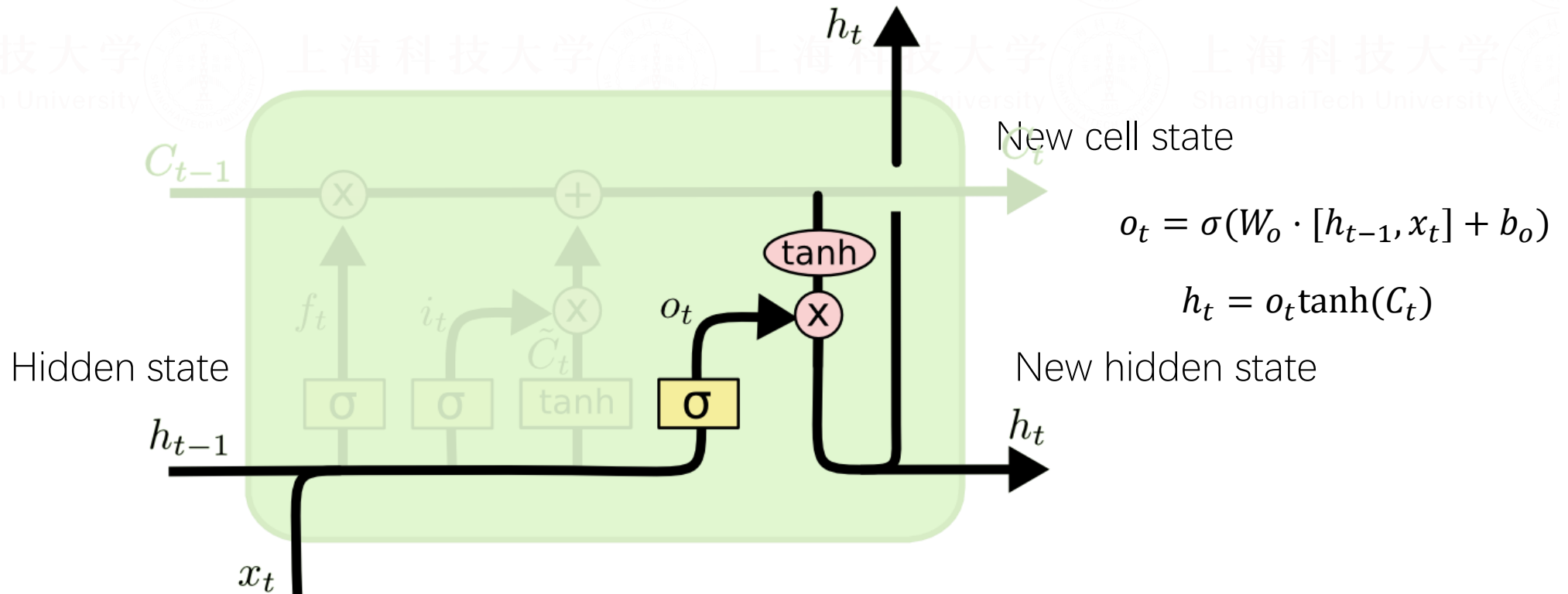
Gating Mechanism – Cell State Update

- A learnable parameter that dynamically decide how many information get through.



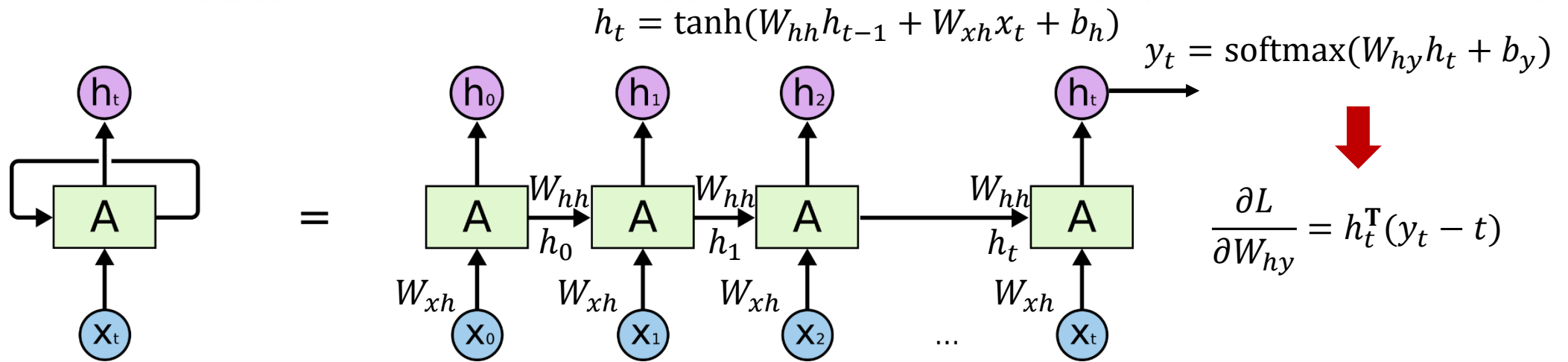
Gating Mechanism – Output Gate

- A learnable parameter that dynamically decide how many information get through.



Recurrent Networks - Backpropagation

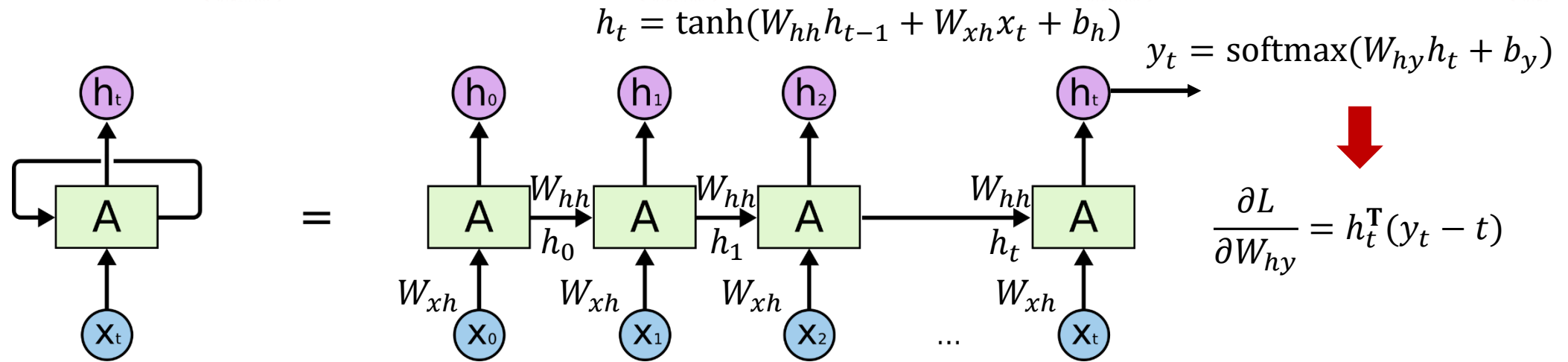
- Backpropagation through time (BPTT)



$$\begin{aligned}
 \frac{\partial L}{\partial W_{xh}} &= \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial W_{xh}} + \frac{\partial L}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial W_{xh}} + \dots + \frac{\partial L}{\partial h_0} \frac{\partial h_0}{\partial W_{xh}} \\
 &= \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial W_{xh}} + \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial W_{xh}} + \dots + \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial h_{t-2}} \dots \frac{\partial h_1}{\partial h_0} \frac{\partial h_0}{\partial W_{xh}}
 \end{aligned}$$

Recurrent Networks – Backpropagation (Cont'd)

- Backpropagation through time (BPTT)



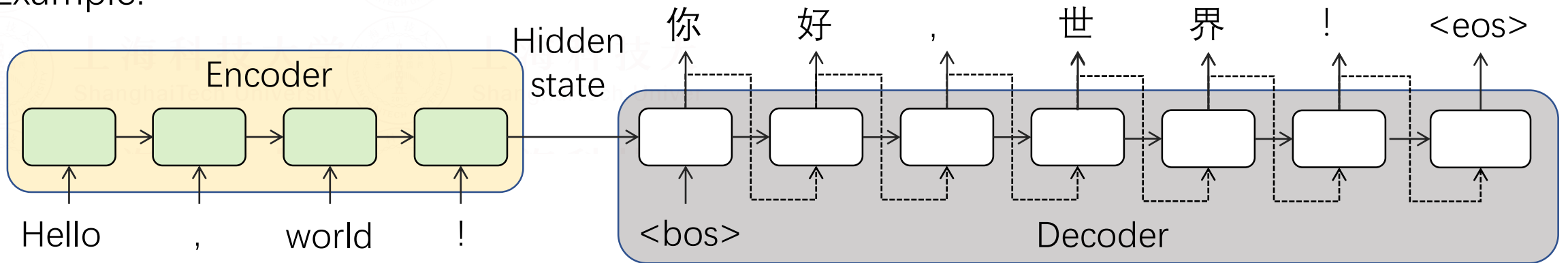
$$\begin{aligned} \frac{\partial L}{\partial W_{hh}} &= \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial W_{hh}} + \frac{\partial L}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial W_{hh}} + \dots + \frac{\partial L}{\partial h_0} \frac{\partial h_0}{\partial W_{hh}} \\ &= \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial W_{hh}} + \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial W_{hh}} + \dots + \frac{\partial L}{\partial h_t} \frac{\partial h_t}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial h_{t-2}} \dots \frac{\partial h_1}{\partial h_0} \frac{\partial h_0}{\partial W_{hh}} \end{aligned}$$

Seq2Seq Tasks

- Encoder-Decoder architecture

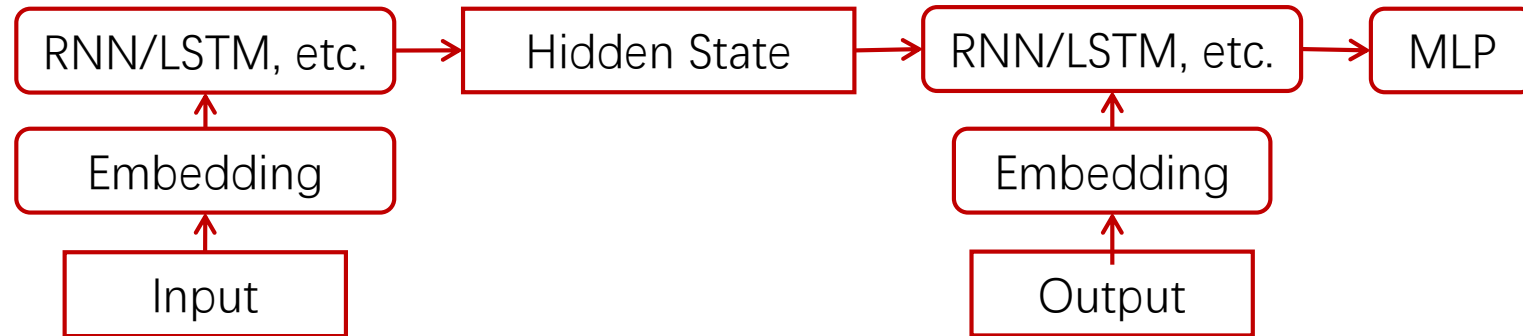


- Machine translation and beyond.
- Example:

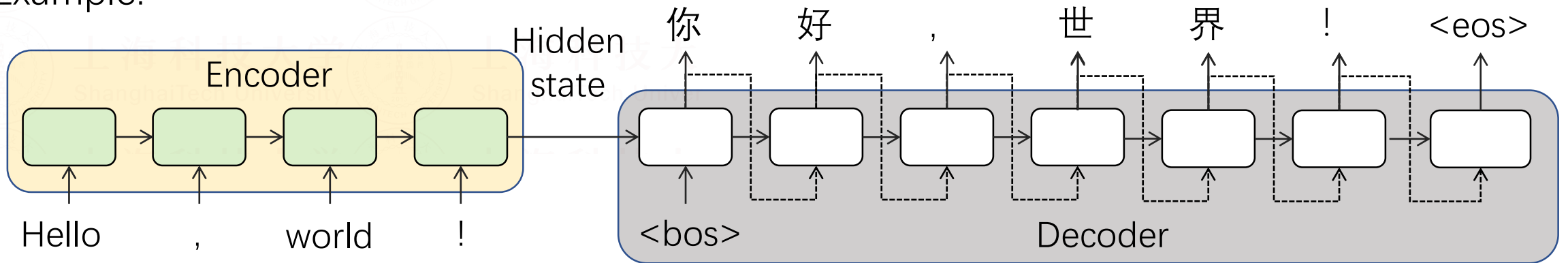


Seq2Seq Tasks (Cont'd)

- Encoder-Decoder architecture



- Machine translation and beyond.
- Example:



Seq2Seq Tasks: Embeddings

- Represent a word (or an item in recommendation system) as a dense vector
- Data compression
 - E.g., word2vec, word embedding (Google 2013)
- Motivation
 - Too many words (20k speech, 500k big vocab), one-hot code extremely sparse
 - One-hot coding ignores meanings/relationship of words (synonym, taxonomy)
 - Create a representation that can represent meaning of words

government debt problems turning into banking crises as has happened

saying that Europe needs unified banking regulation to replace the hodgepodge



Embeddings

$$J_-(\theta) = \prod_{t=1}^T \prod_{\substack{-m \leq j \leq m \\ j \neq 0}} P(w_{t+j} | w_t; \theta)$$

w_{t-2} w_{t-1} w_t w_{t+1} w_{t+2}
turning into **banking** crises as
 $\underbrace{P(w_{t-1} | w_t)}_{P(w_{t-2} | w_t)}$ $\underbrace{P(w_{t+1} | w_t)}_{\dots}$

or negative log likelihood $J(\theta) = -\frac{1}{T} \sum_{t=1}^T \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \log P(w_{t+j} | w_t; \theta)$

$$P(w_{t+j} | w_t) = \frac{\exp(u_{w_{t+j}}^T v_{w_t})}{\sum \exp(u_{w_v}^T v_{w_t})}$$

Word vector as
a center word

Word vector as
a context word

Embeddings (Cont'd)

$$P(w_{t+j}|w_t) = \frac{\exp(u_{w_{t+j}}^T v_{w_t})}{\sum \exp(u_{w_v}^T v_{w_t})}$$

One-hot code
for V words

1*V

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}^T$$

$$\min J(\theta) = \prod_{t=1}^T \prod_{\substack{-m \leq j \leq m \\ j \neq 0}} P(w_{t+j}|w_t; \theta)$$

$$\begin{bmatrix} 0.1 & 0.2 & -0.3 & 0.4 & -0.5 & 0.6 & 0.1 & 0 \end{bmatrix}$$

matrix, as center word
V*d

Can be the same
in modern LLMs



1*d

$$\begin{bmatrix} u & \dots & u \\ \vdots & \ddots & \vdots \\ u & \dots & u \end{bmatrix}$$

matrix, as context word
d*V



$$\begin{bmatrix} 0.1 \\ -0.9 \\ 0.2 \\ \vdots \\ \vdots \\ \vdots \end{bmatrix}^T$$

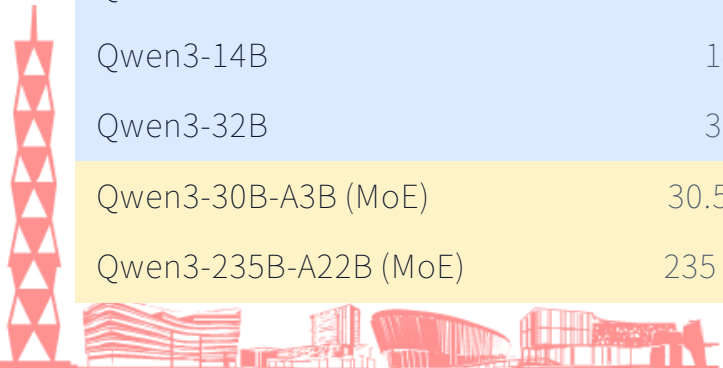
softmax

$$\begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ \vdots \\ \vdots \\ \vdots \end{bmatrix}^T$$

w_{t-2} w_{t-1} w_t w_{t+1} w_{t+2}
turning into **banking** crises as
 $P(w_{t-2}|w_t)$ $P(w_{t-1}|w_t)$ $P(w_{t+1}|w_t)$...

Embedding Table in Qwen LLMs

Model	Size of the Model	Vocab Size	Hidden Size	Embedding Params.
Qwen2.5-0.5B	0.49 B	151,936	896	136.1 M
Qwen2.5-1.5B	1.54 B	151,936	1,536	233.4 M
Qwen2.5-3B	3.09 B	151,936	2,048	311.2 M
Qwen2.5-7B	7.62 B	152,064	3,584	1,090.0 M (Untied)
Qwen2.5-14B	14.7 B	152,064	5,120	1,557.1 M (Untied)
Qwen2.5-32B	32.5 B	152,064	5,120	1,557.1 M (Untied)
Qwen2.5-72B	72.7 B	152,064	8,192	2,491.4 M (Untied)
Qwen3-0.6B	0.6 B	151,936	1,024	155.6 M
Qwen3-1.7B	1.7 B	151,936	2,048	311.2 M
Qwen3-4B	4.0 B	151,936	2,560	389.0 M
Qwen3-8B	8.2 B	151,936	4,096	1,244.7 M (Untied)
Qwen3-14B	14.8 B	151,936	5,120	1,555.8 M (Untied)
Qwen3-32B	32.8 B	151,936	5,120	1,555.8 M (Untied)
Qwen3-30B-A3B (MoE)	30.5 B / 3 B	151,936	2,048	622.3 M (Untied)
Qwen3-235B-A22B (MoE)	235 B / 22 B	151,936	4,096	1,244.7 M (Untied)

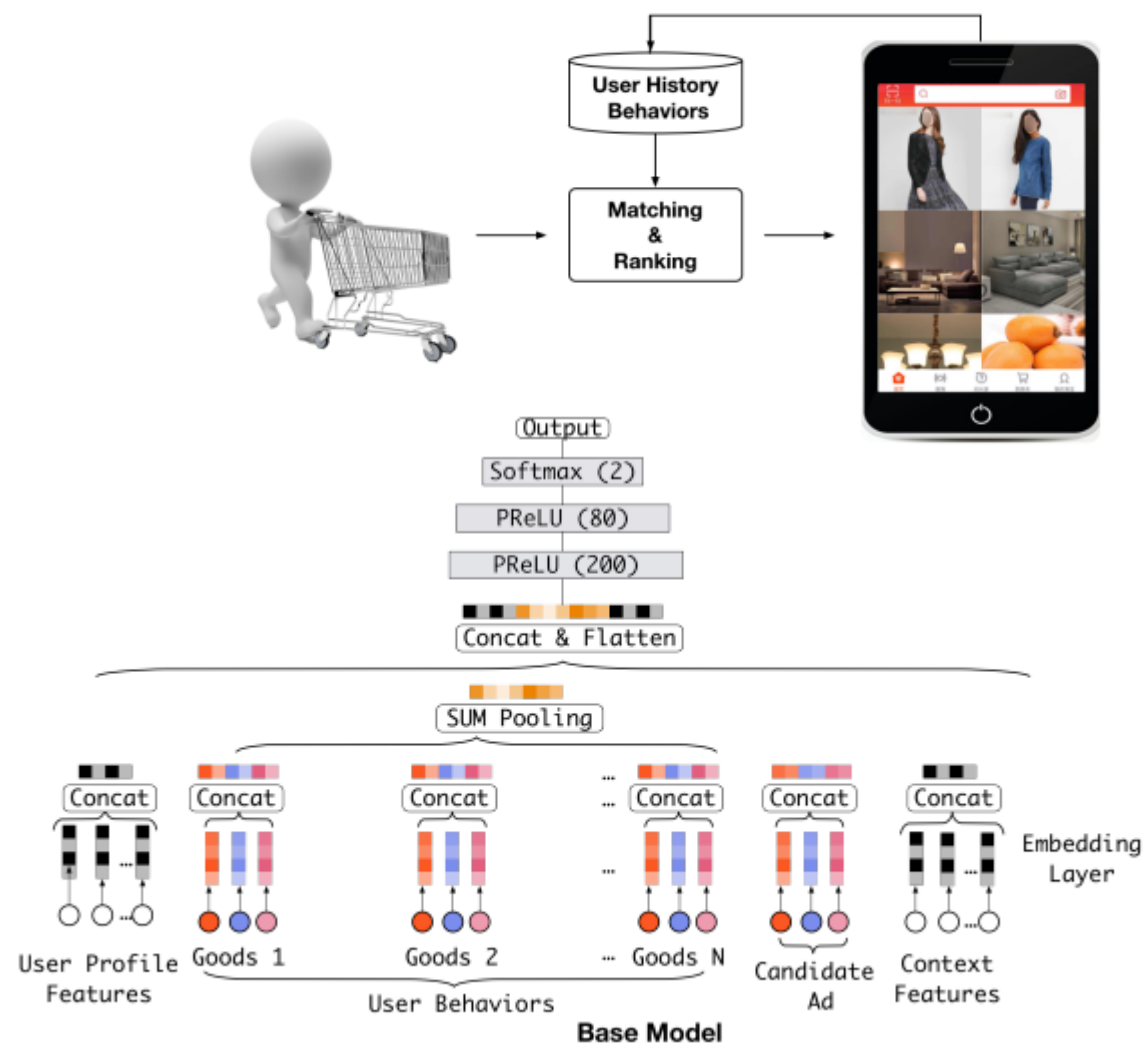


Embeddings in Recommendation Systems

- Recommendation, click-through rate (CTR) prediction
- User/ads features, highly sparse, use embeddings

$[0,0,0,0,1,0,0]$ Weekday = Friday
 $[0,1]$ Gender = Female
 $[0,...,1,...,1,...,0]$ Visited_cate_id = [Bag, Book]
 $[0,...,1,...,0]$ Ad_cate_id = [Book]

Category	Feature Group	Dimemsionality	Type	#Nonzero Ids per Instance
User Profile Features	gender	2	one-hot	1
	age_level	~ 10	one-hot	1
	...	...	...	...
User Behavior Features	visited_goods_ids	$\sim 10^9$	multi-hot	$\sim 10^3$
	visited_shop_ids	$\sim 10^7$	multi-hot	$\sim 10^3$
	visited_cate_ids	$\sim 10^4$	multi-hot	$\sim 10^2$
Ad Features	goods_id	$\sim 10^7$	one-hot	1
	shop_id	$\sim 10^5$	one-hot	1
	cate_id	$\sim 10^4$	one-hot	1
	...	...	...	...
Context Features	pid	~ 10	one-hot	1
	time	~ 10	one-hot	1
	...	...	...	...

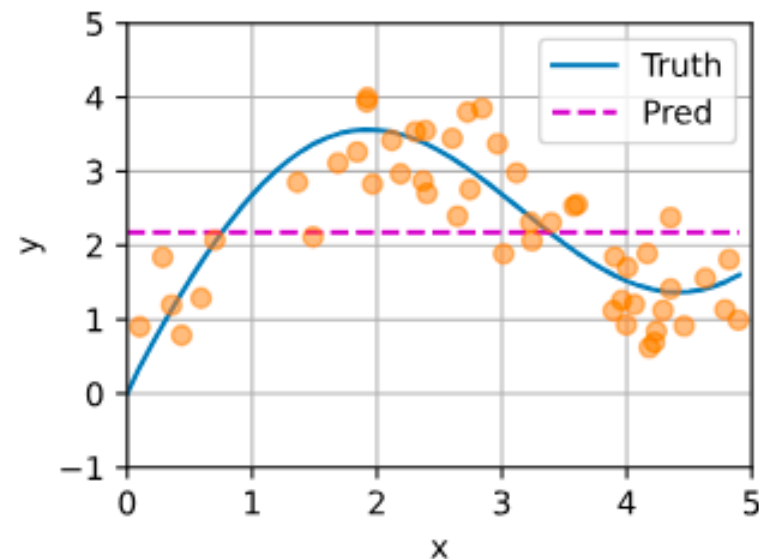


Attention Mechanism

$$f(x) = \sum_{i=1}^n \alpha(x, x_i) y_i$$

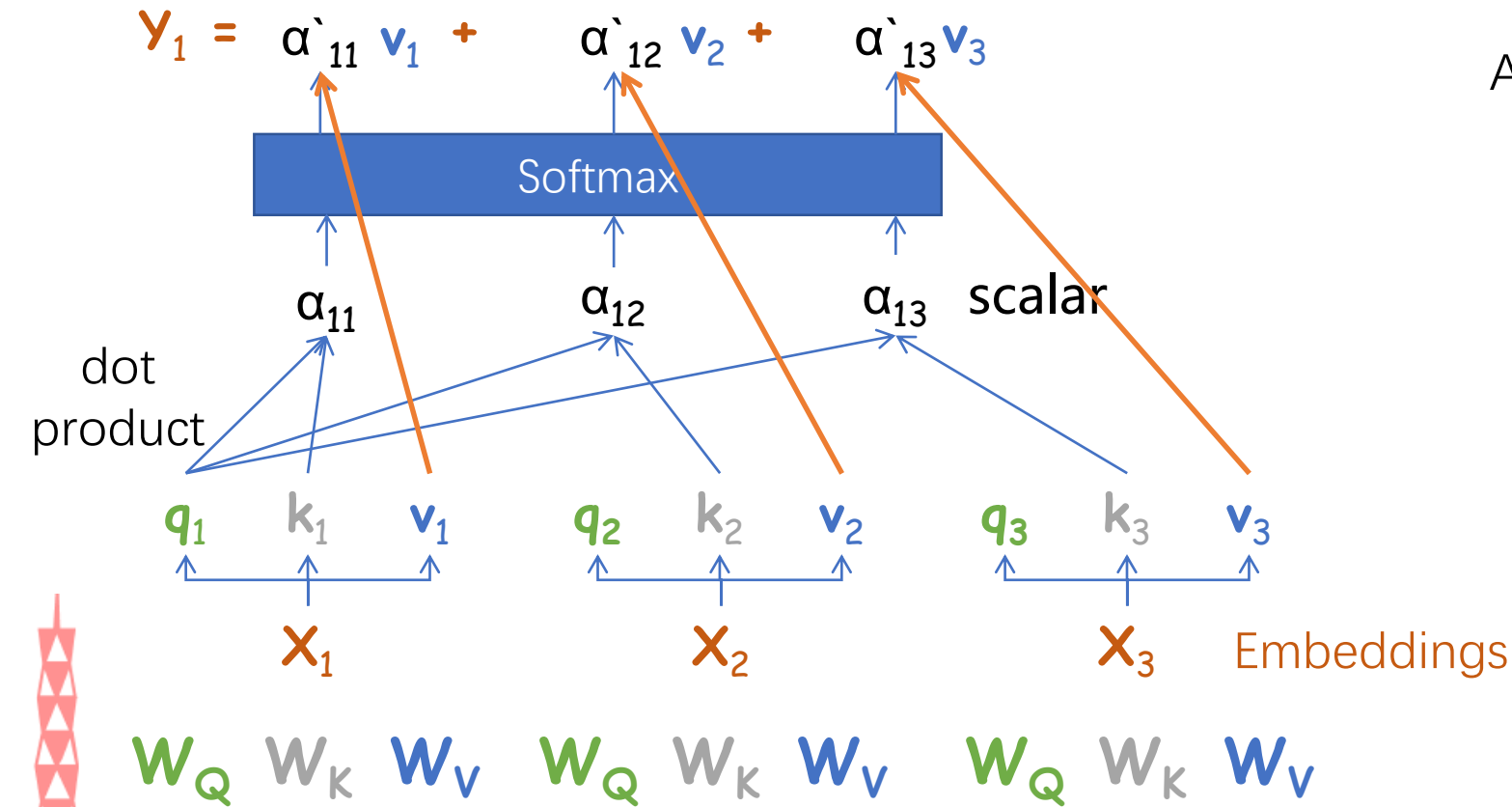
Diagram illustrating the Attention Mechanism formula:

- Query**: Points to x in the formula.
- Key**: Points to x_i in the formula.
- Value**: Points to y_i in the formula.
- Attention score**: Points to $\alpha(x, x_i)$ in the formula.

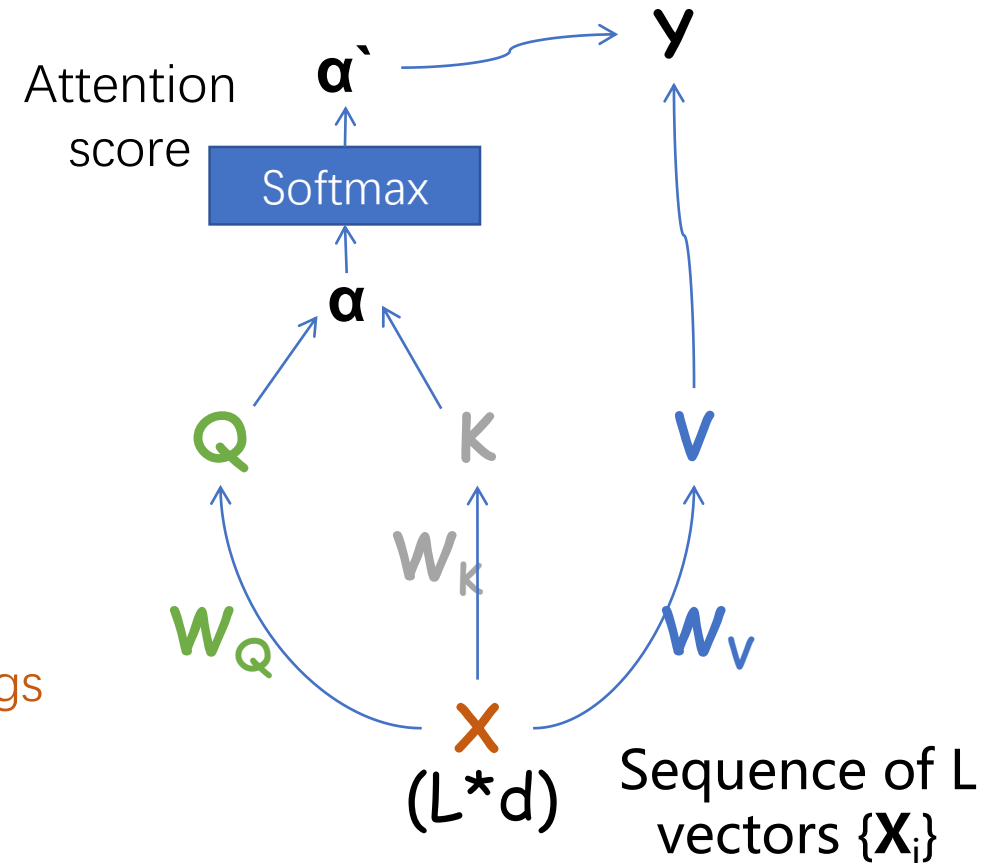


Self-Attention

- Q, K, V are generated by the inputs themselves

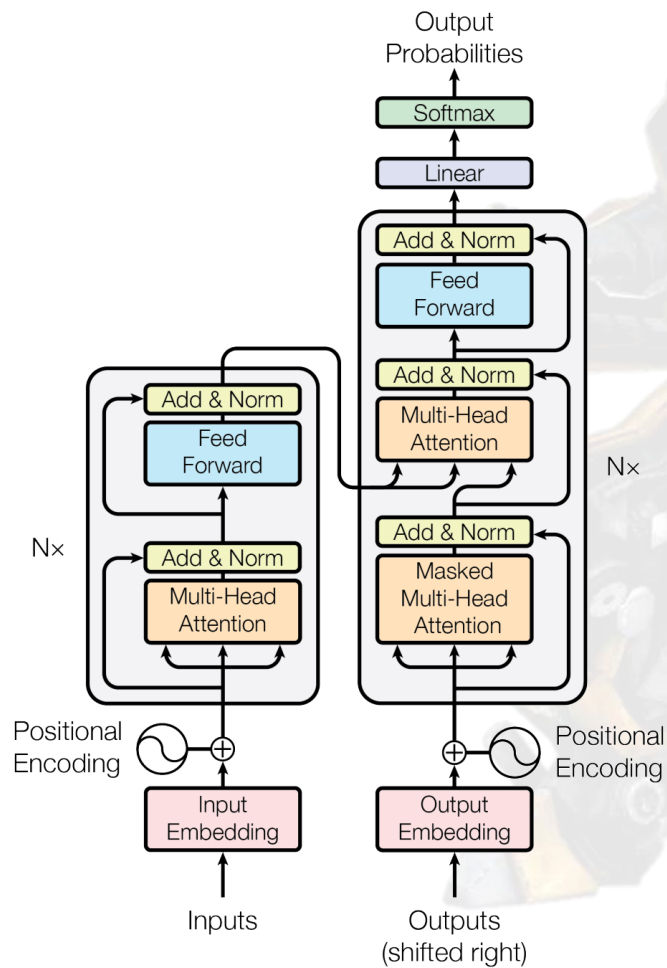


A simplified self-attention layer

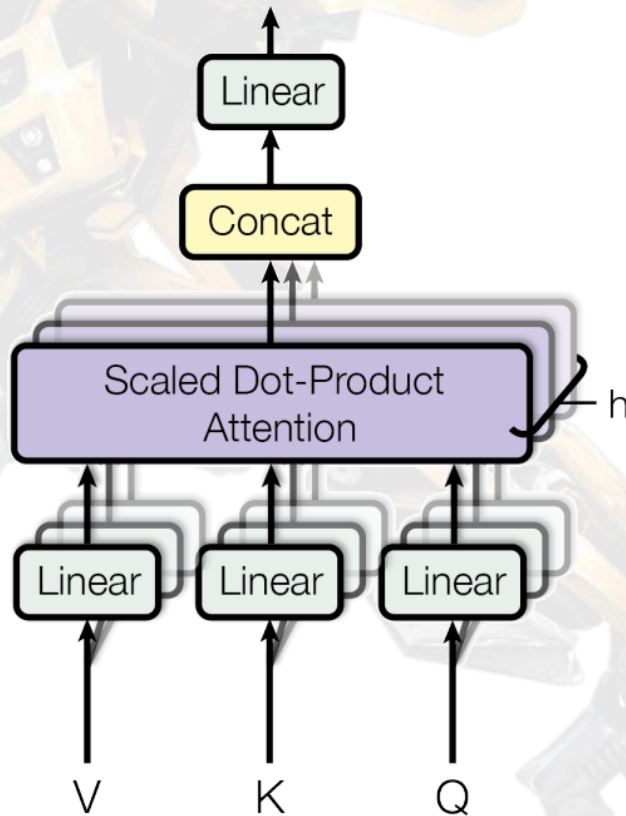


Transformer

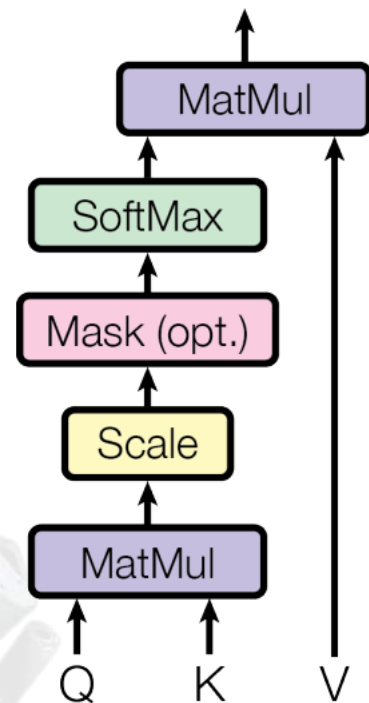
- Attention is all you need



Multi-Head Attention



Scaled Dot-Product Attention



$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Copyright: <https://www.fanpop.com/>

<https://arxiv.org/pdf/1706.03762.pdf>

立志成才 报国裕民

Transformer (Cont'd)

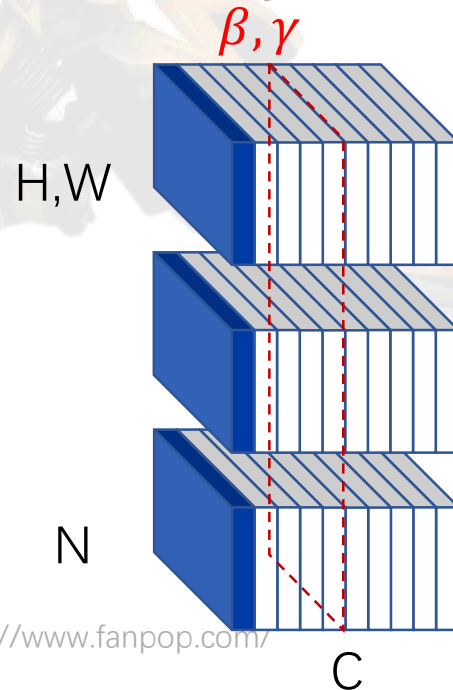
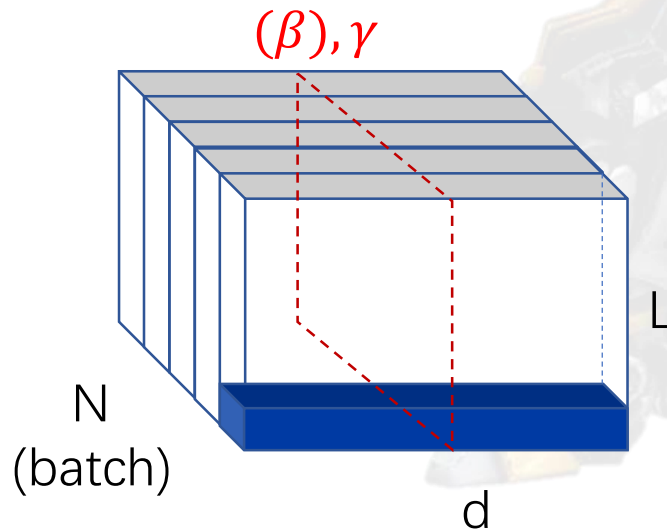
- Positional encoding

$$P \in R^{n \times d}: p_{i,2j} = \sin\left(\frac{i}{10000^{2j/d}}\right),$$

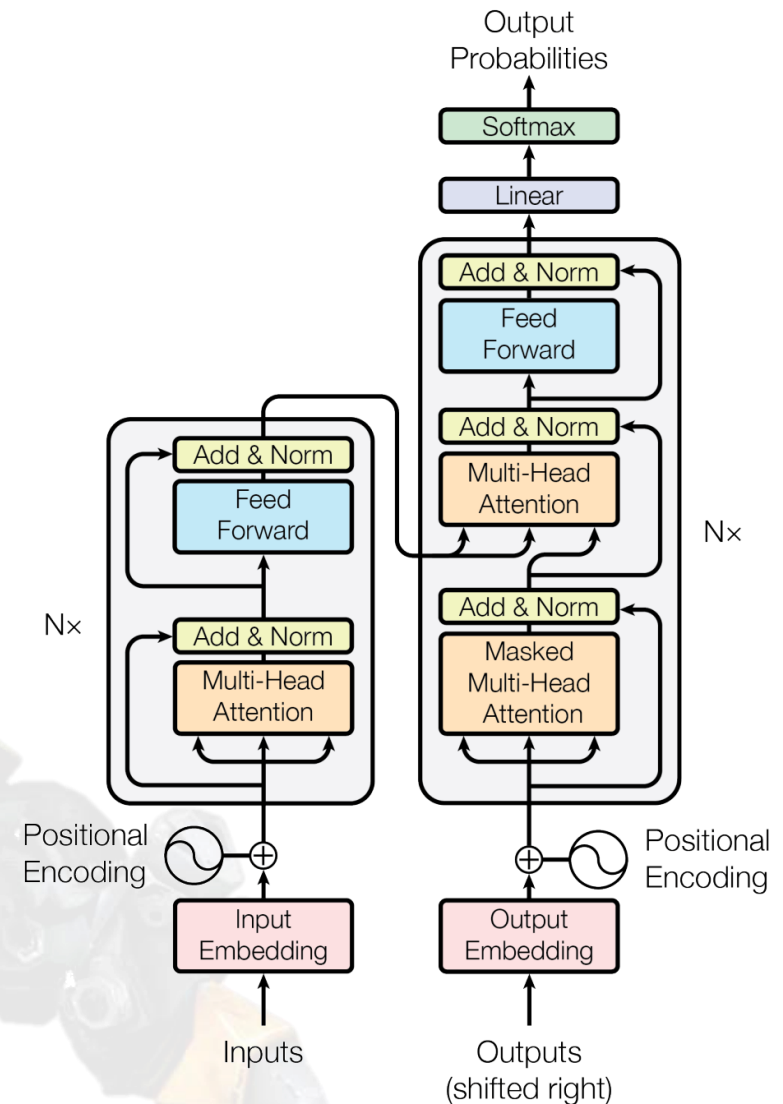
$$p_{i,2j+1} = \cos\left(\frac{i}{10000^{2j/d}}\right)$$

0	in	binary	is	000
1	in	binary	is	001
2	in	binary	is	010
3	in	binary	is	011
4	in	binary	is	100
5	in	binary	is	101
6	in	binary	is	110
7	in	binary	is	111

- Layer vs. Batch norm.

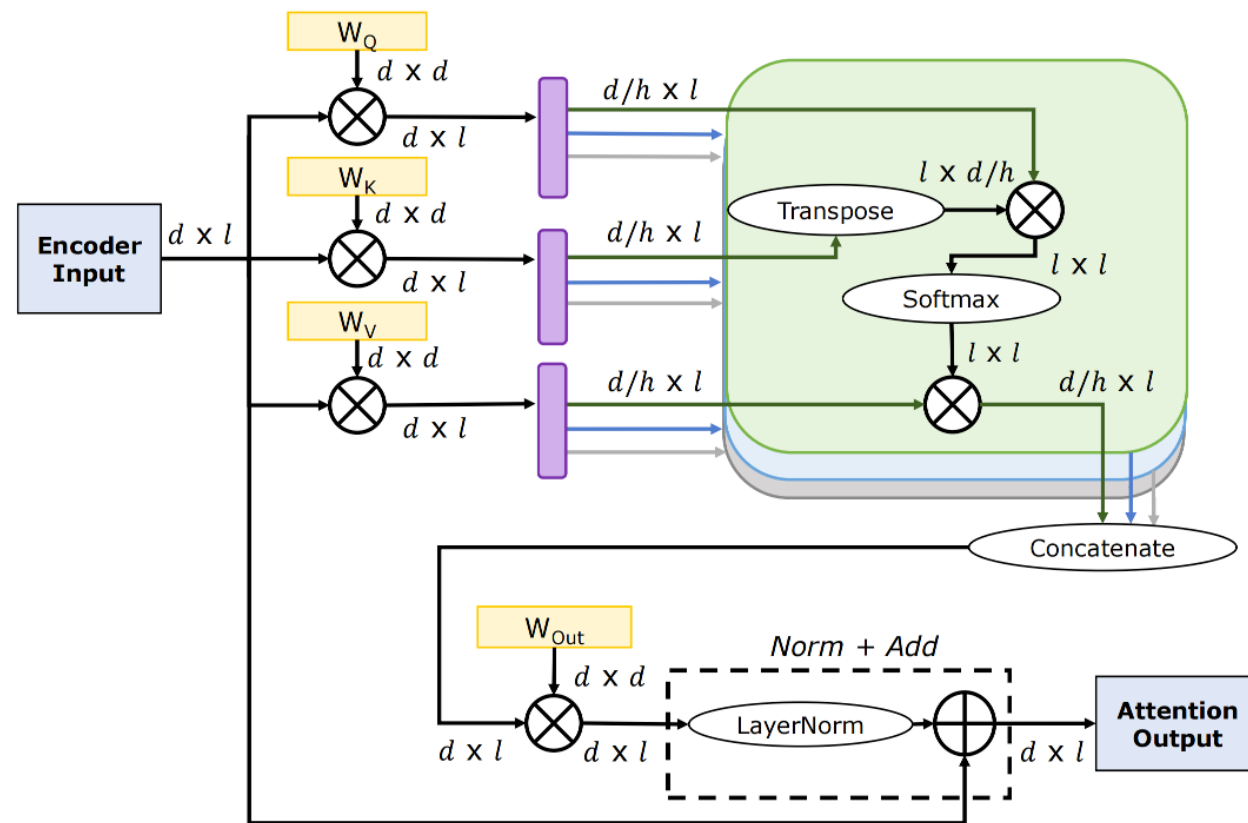
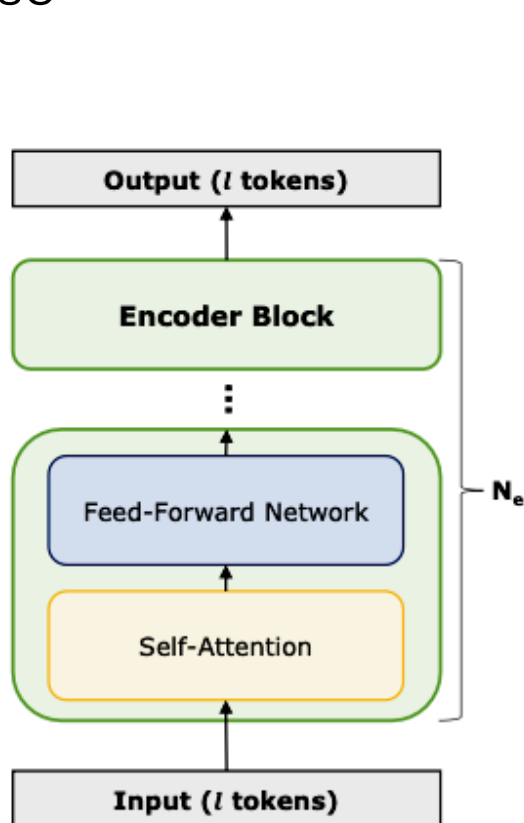


Copyright: <https://www.fanpop.com/>



Encoder-only Models

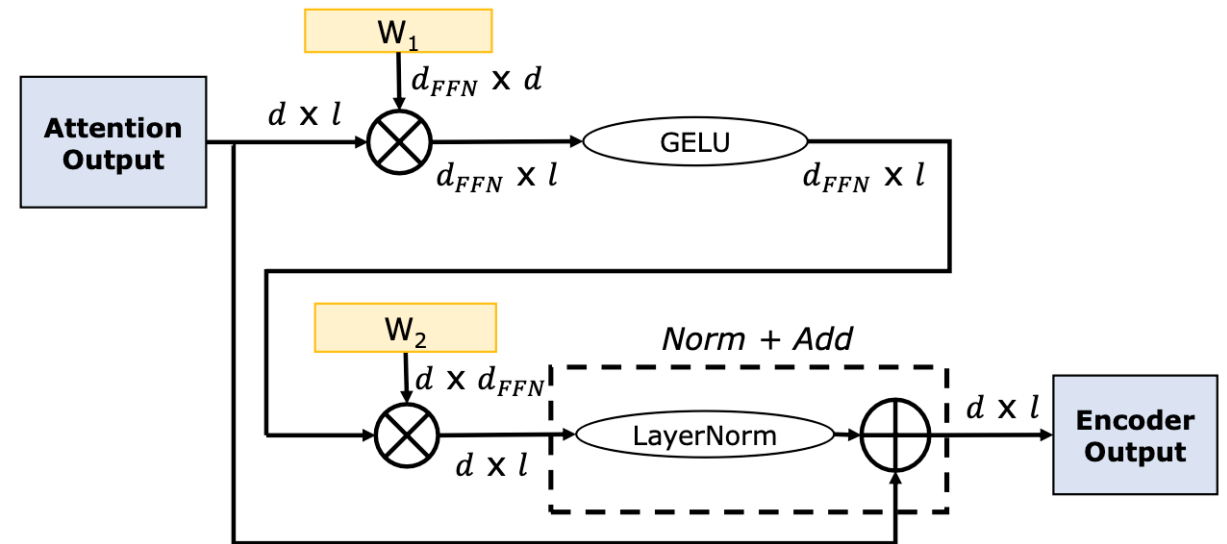
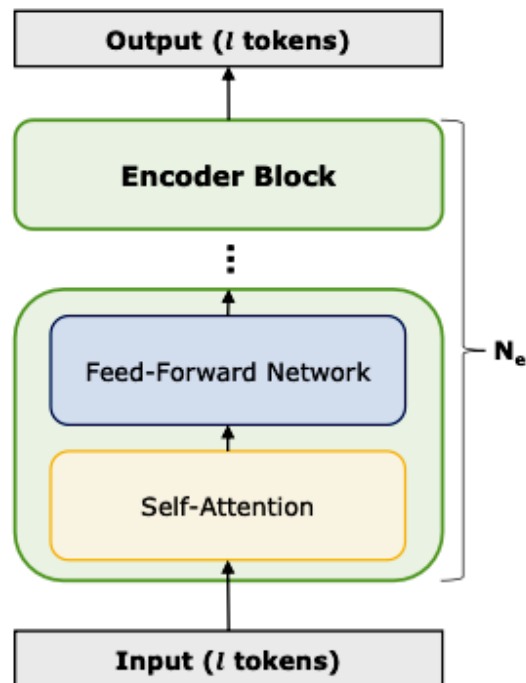
- BERT-base



Muti-Head Attention (MHA) Module

Encoder-only Models (Cont'd)

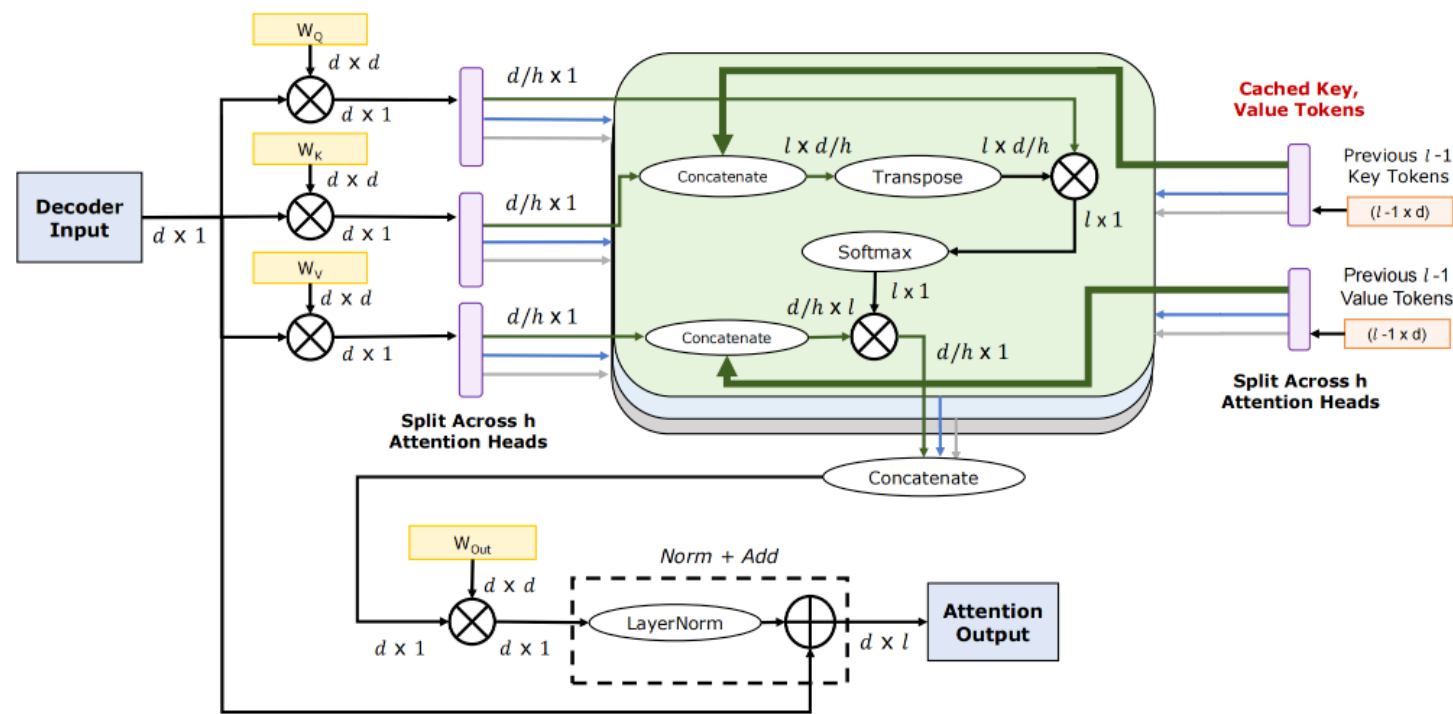
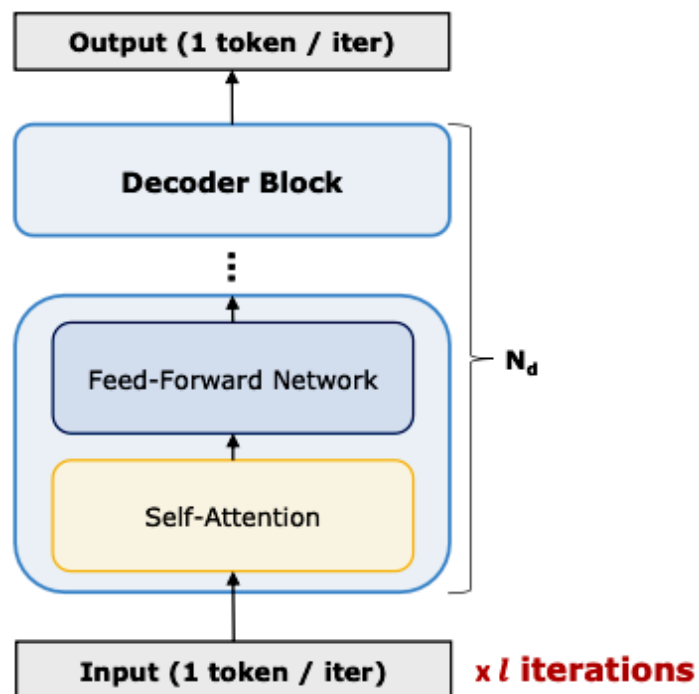
- BERT-base



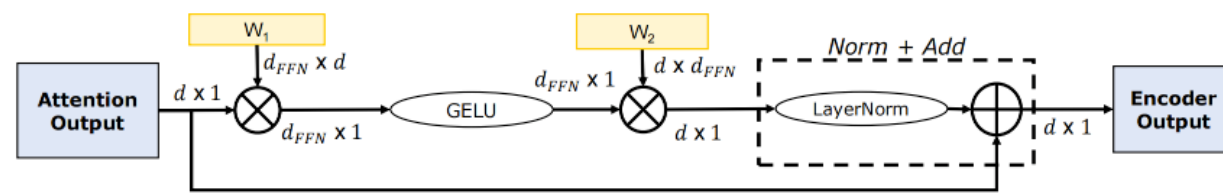
Feed-Forward Network (FFN) Module

Decoder-only Models

- GPT-2



Multi-Head Attention (MHA) Module



Feed-Forward Network (FFN) Module

Further Reading

- Simon O. Haykin, Neural Networks and Learning Machines, third edition.
- Deep Learning. <https://www.deeplearningbook.org/>. Ian Goodfellow and Yoshua Bengio and Aaron Courville, an MIT Press book.
- 智能计算系统. 陈云霁等编著
- [Stanford University CS231n: Deep Learning for Computer Vision](#)
- APS360 Fundamentals of AI.
https://www.cs.toronto.edu/~lczhang/aps360_20191/
- https://zh-v2.d2l.ai/chapter_attention-mechanisms/index.html
- The other NN structures (GAN, GNN, and beyond)
- *Full Stack Optimization of Transformer Inference: a Survey*



MQA/GQA/MLA

- Multi-query attention
 - h heads share the same KV, KV size reduced by h ;
 - Quality degraded;
- Grouped-query attention (adopted in Qwen2, Llama 3, etc.)
 - g groups share the same KV, KV size reduced to g/h ;
 - Quality slightly degraded;
- Multi-Latent attention: low-rank approximation

