

# AI by Hand



## Volume 1

Basic x 5  
Advanced x 20

Prof. Tom Yeh

2024

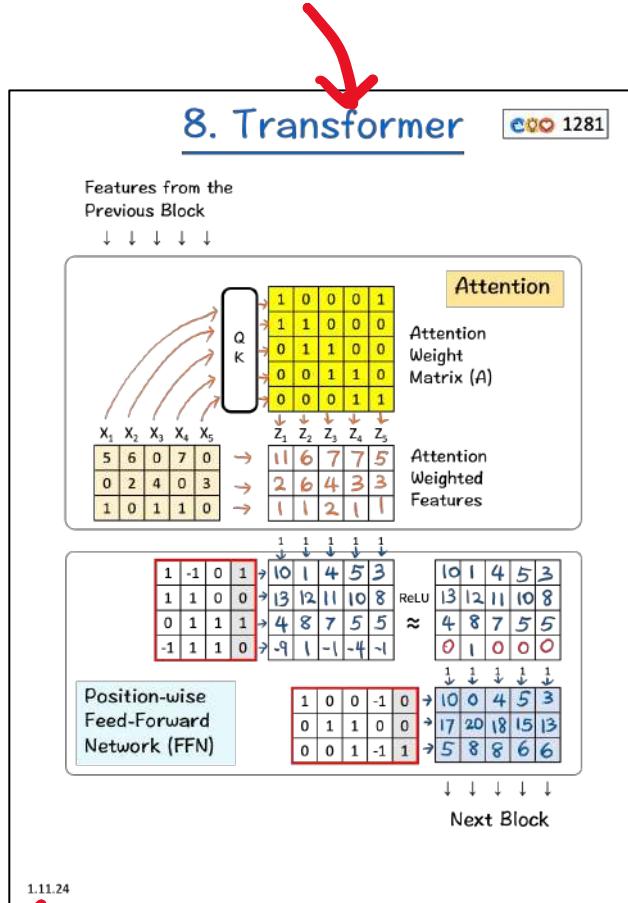
## Basic

- I. One Node
- II. Four Nodes
- III. One Hidden Layer
- IV. Three Inputs
- V. Seven Layers

## Advanced

- 1. Mixture of Experts (MOEs)
- 2. Recurrent Neural Network (RNN)
- 3. Mamba
- 4. Matrix Multiplication
- 5. LLM Sampling
- 6. MLP in PyTorch
- 7. Backpropagation
- 8. Transformer
- 9. Batch Normalization
- 10. Generative Adversarial Network (GAN)
- 11. Self Attention
- 12. Dropout
- 13. Autoencoder
- 14. Vector Database
- 15. CLIP
- 16. Residual Network (ResNet)
- 17. Graph Convolution Network (GCN)
- 18. SORA's Diffusion Transformer (DiT)
- 19. Gemini 1.5's Switch Transformer
- 20. Reinforcement Learning with Human Feedback (RLHF)

Link to my original LinkedIn post  
with animation and explanation



Date originally posted

# I. One Node



$$\begin{array}{c} \boxed{2} \\ \boxed{1} \\ \boxed{3} \end{array} \times \text{ReLU}$$

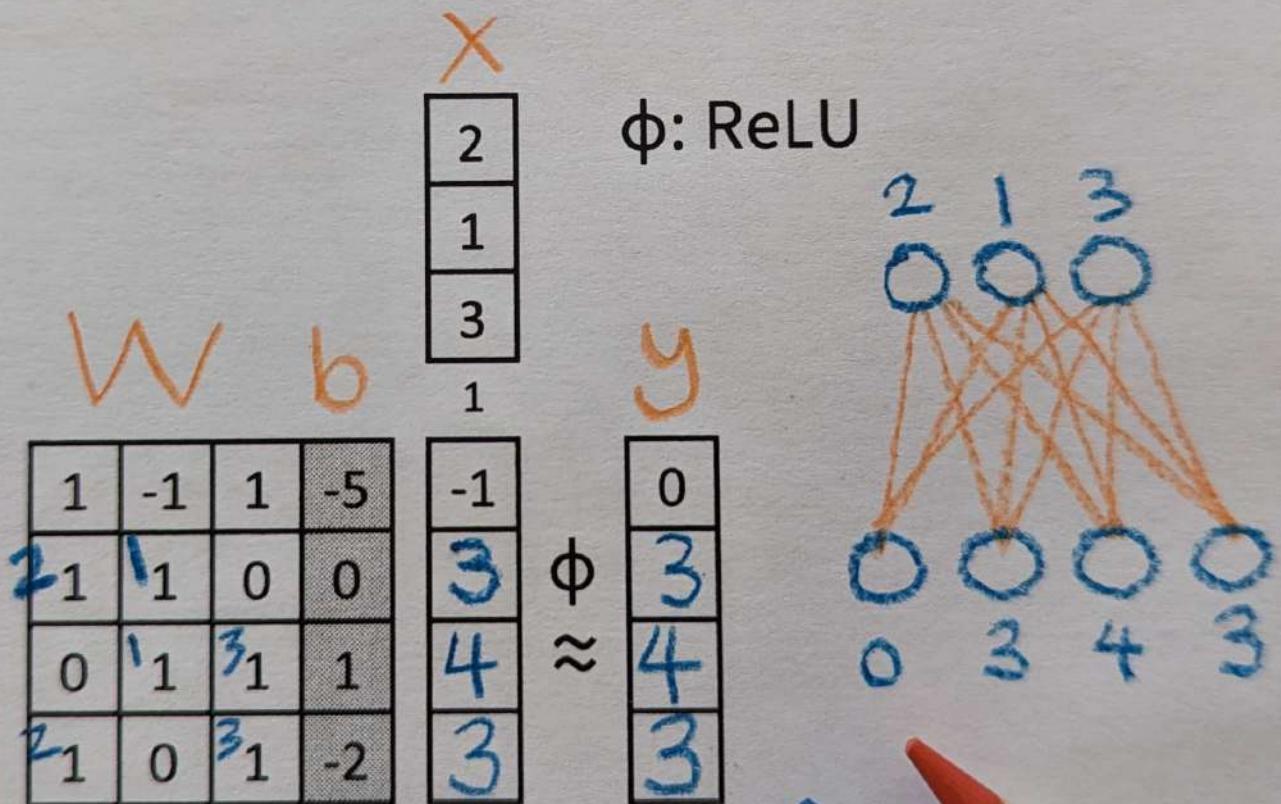
$w$        $b$        $1$        $\phi$        $\approx$        $0$

$$\begin{array}{|c|c|c|c|} \hline 1 & -1 & 1 & -5 \\ \hline \end{array} \quad \boxed{-1}$$



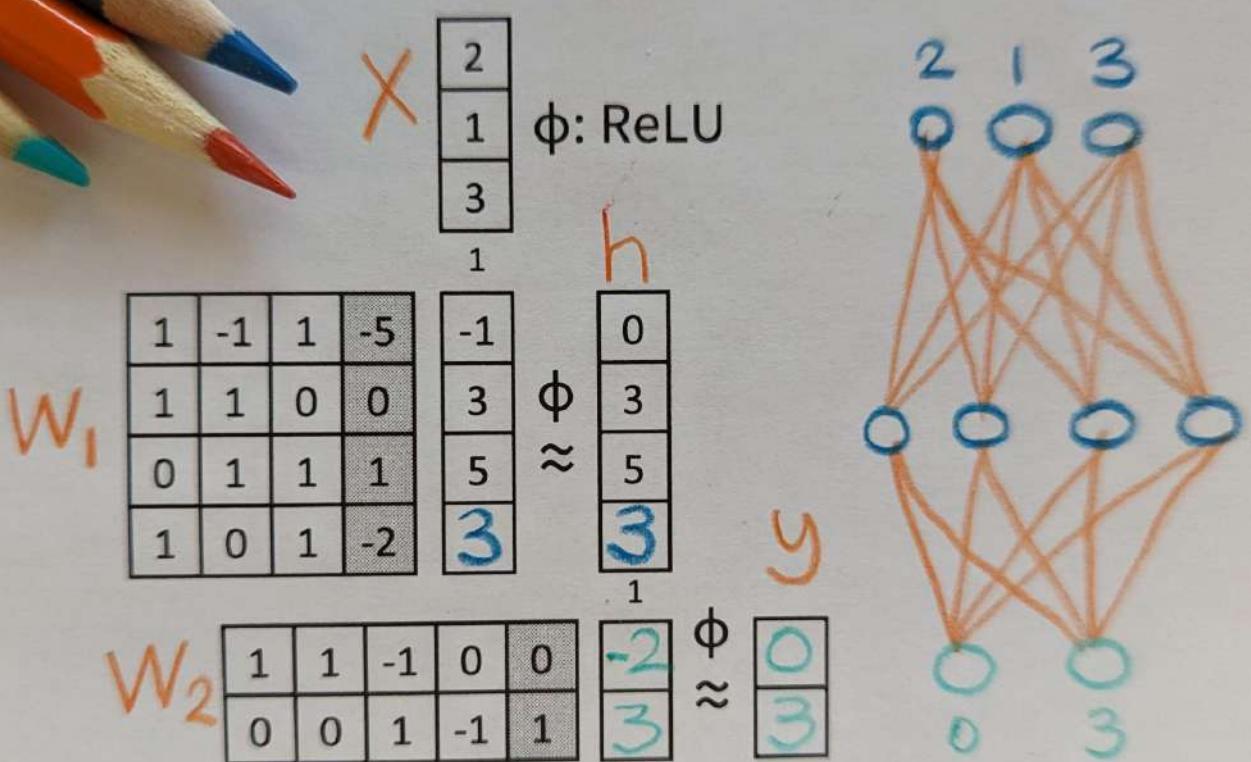
© 2023 Tom Yeh.

## II. Four Nodes



© 2023 Tom Yeh.

### III. Hidden Layer



© 2023 Tom Yeh.

## IV. Three Inputs

2	1	0
1	1	1
3	0	1
1	1	1

$\phi$ : ReLU

1	-1	1	-5
1	1	0	0
0	1	1	1
1	0	1	-2

-1	-5	-5
3	2	1
5	2	3
3	-1	-1

$\phi \approx$

0	0	0
3	2	1
5	2	3
3	0	0

1 1 1

1	1	-1	0	0
0	0	1	-1	1

-2	0	-2
3	3	4

$\phi \approx$

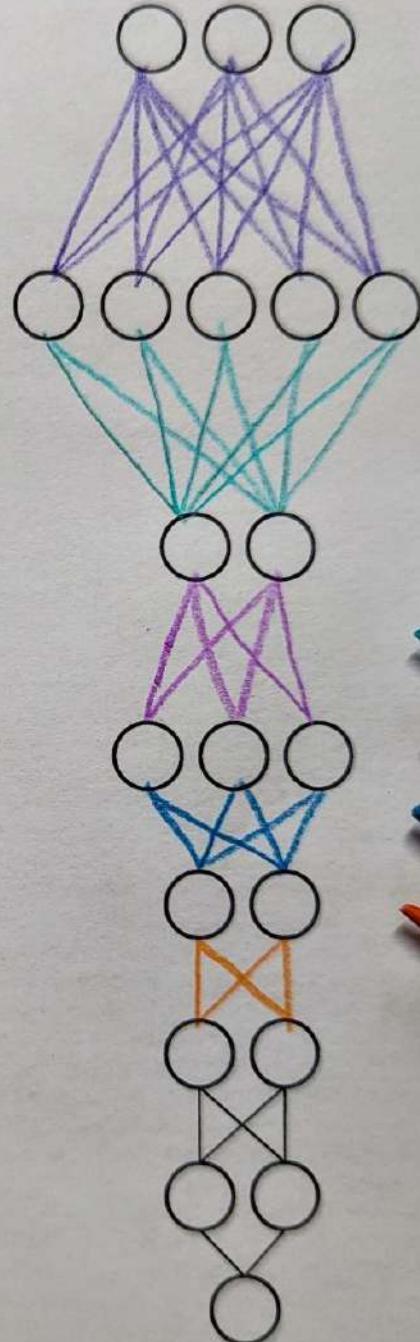
0	0	0
3	3	4



© 2023 Tom Yeh.

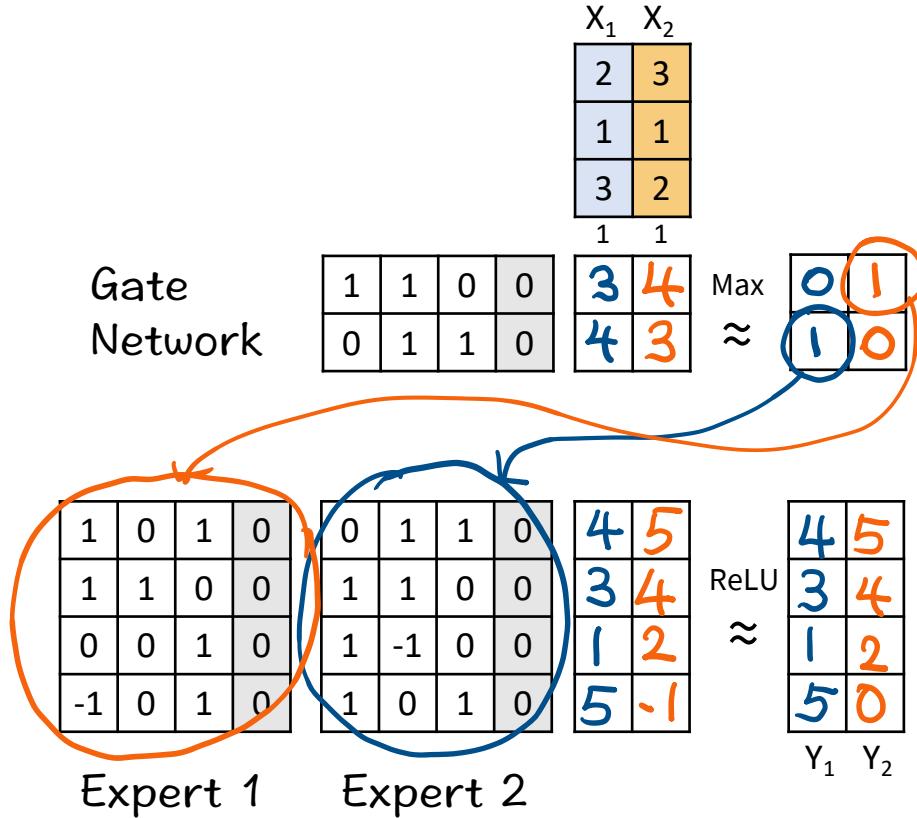
# V. Seven Layers

					3 5
					4 4
					5 3
	0 0 1				5 3
$w_1$	0 1 0				4 4
	1 0 0				3 5
	1 1 0				7 9
	0 1 1				9 7
$w_2$	1 1 -1 0 0				6 2
	0 0 1 1 -1				1 7
$w_3$	1 1				7 9
	1 -1				5 <del>5</del>
	1 2				8 16
$w_4$	1 -1 0				2 9
	0 -1 1				3 16
$w_5$	0 1				3 21
	1 0				2 9
$w_6$	1 -1				1 12
	1 1				5 30
$w_7$	1 -1		-4	-18	



© 2023 Tom Yeh.

# 1. Mixture of Experts



## 2. Recurrent Neural Network (RNN)



Input Sequence

X	3	4	5	6
---	---	---	---	---

Parameters

$$A \begin{array}{|c|c|} \hline 1 & -1 \\ \hline 1 & 1 \\ \hline \end{array} \quad B \begin{array}{|c|c|} \hline 1 \\ \hline 2 \\ \hline \end{array} \quad C \begin{array}{|c|c|} \hline -1 & 1 \\ \hline \end{array}$$

Activation Function

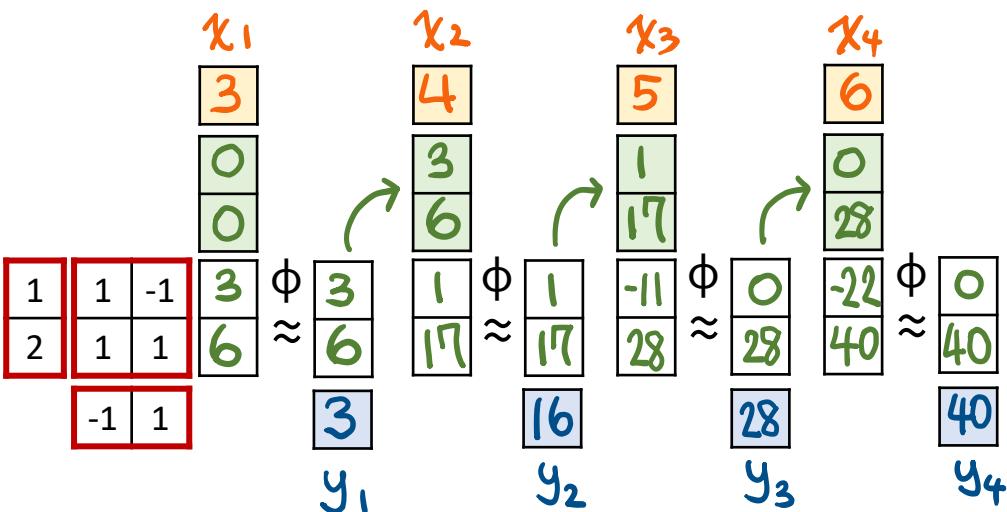
$\phi$ : ReLU

Hidden States

$H_0$	0	
	0	

Output Sequence

y				
---	--	--	--	--

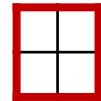


### 3. Mamba's S6 Model

Input Sequence

3	4	5	6
---	---	---	---

Parameters

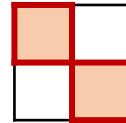


Output Sequence

-1	2	-3	24	1
----	---	----	----	---

Selective

Structured



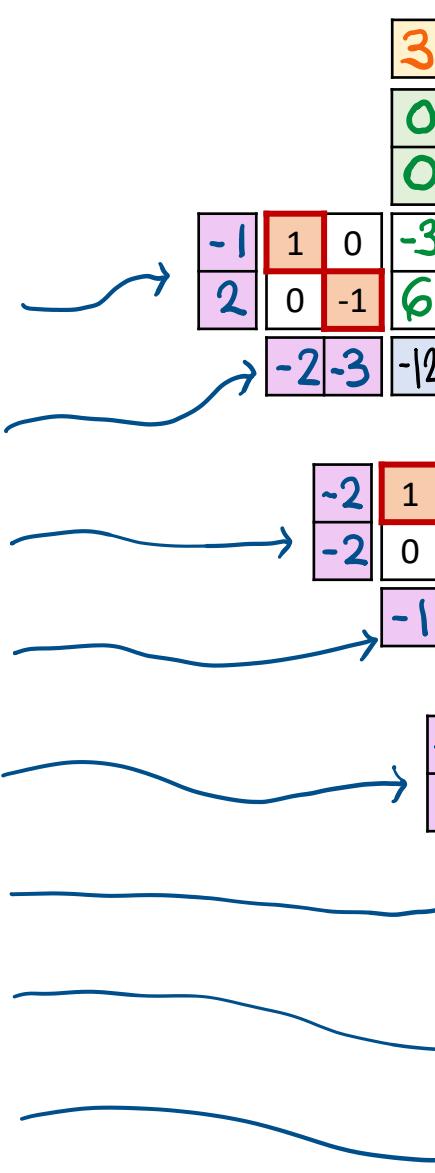
State-Space



1	-1	0	0
0	-1	0	1
1	0	-1	0
1	0	0	-1
1	0	-1	0
0	1	0	-1
1	-1	0	0
0	0	-1	1
-1	0	0	0
1	0	0	0
0	0	-1	0
0	1	0	0
1	-1	0	0
0	0	-1	1
1	0	0	0
0	-1	1	0

3  
4  
5  
6

-1  
2  
-2  
-3  
-2  
-2  
-1  
1  
-3  
3  
-5  
4  
-1  
1  
3  
1



Scan

-3	-1	0	-4
3	0	1	1
-5	4	24	4
-1	-1	0	-2
1	1	1	7
3	1	1	1

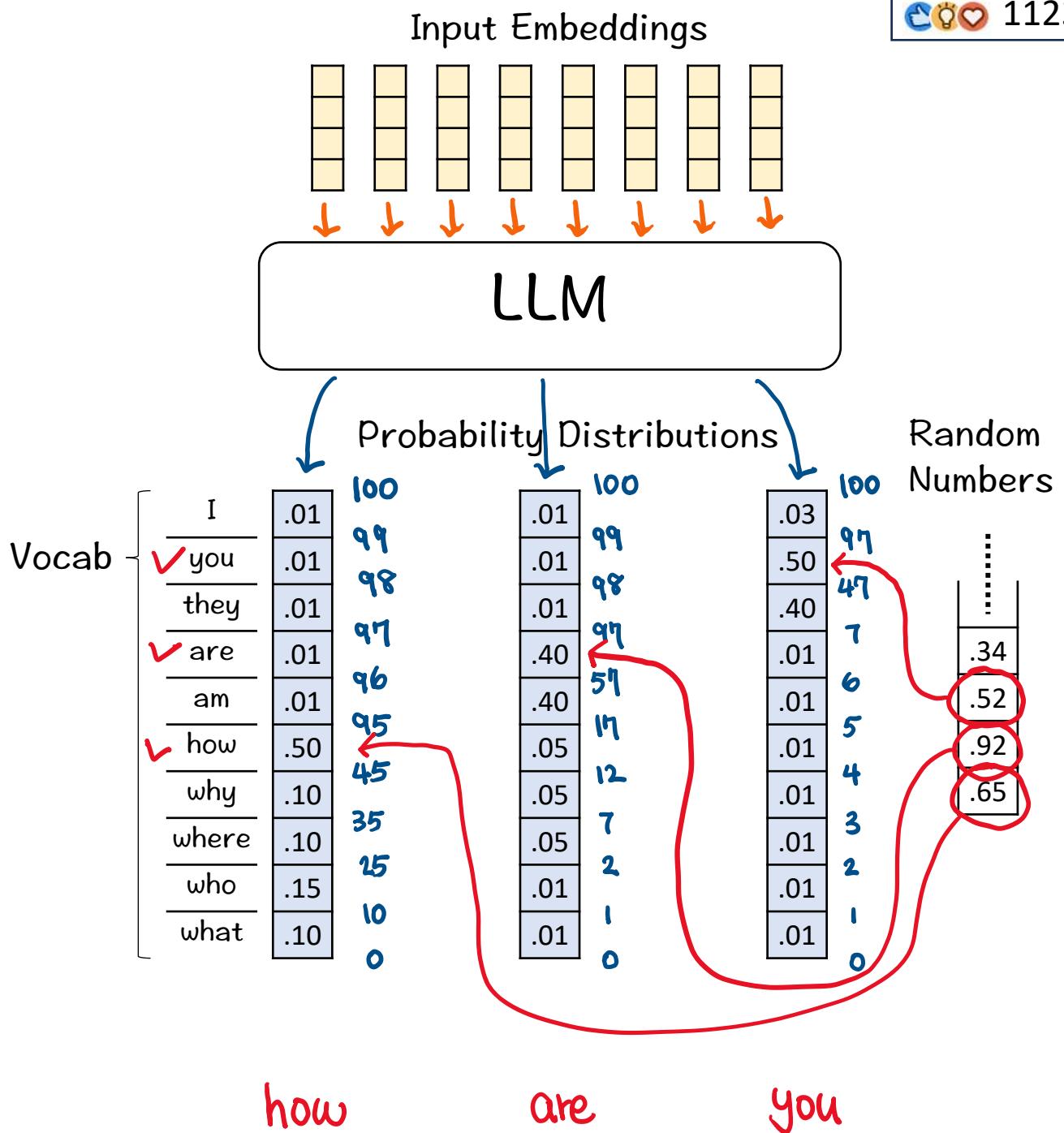
## 4. Matrix Multiplication

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 5 & 2 \\ 2 & 4 & 2 \end{bmatrix} = ?$$

1	5	2
2	4	2

3	9	4
1	-1	0

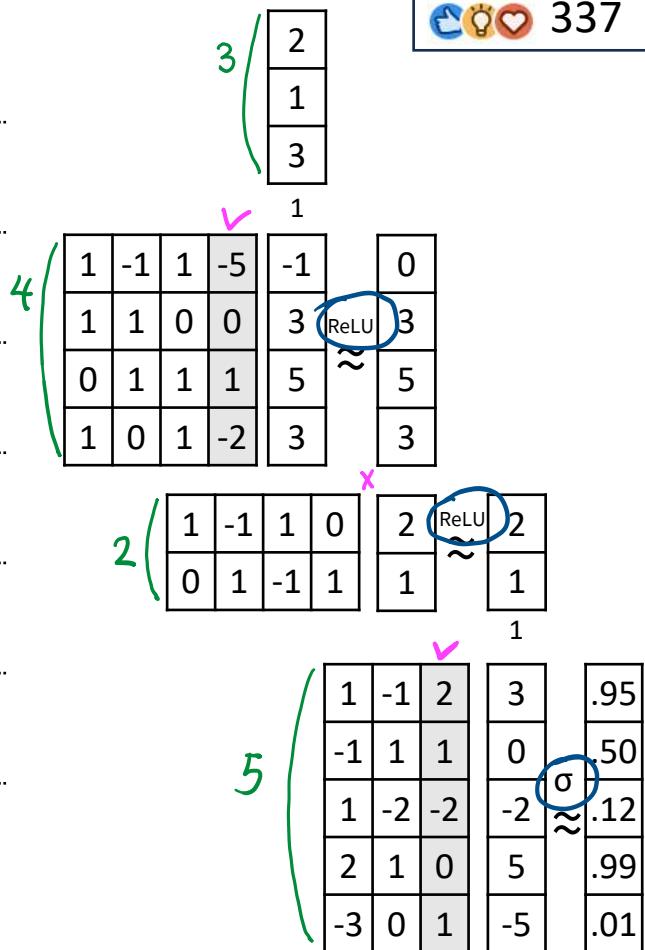
## 5. How does an LLM sample a sentence?



# 6. Multi Layer Perceptron in pytorch



```
1 mlp_model = nn.Sequential(  
.....  
2     nn.Linear( 3, 4, bias = T ),  
.....  
3     nn.ReLU(),  
.....  
4     nn.Linear( 4, 2, bias = F ),  
.....  
5     nn.ReLU(),  
.....  
6     nn.Linear( 2, 5, bias = T ),  
.....  
7     nn.Sigmoid()  
.....  
8 )
```



Hints:

Linear Layer: { Identity | Linear | Bilinear }

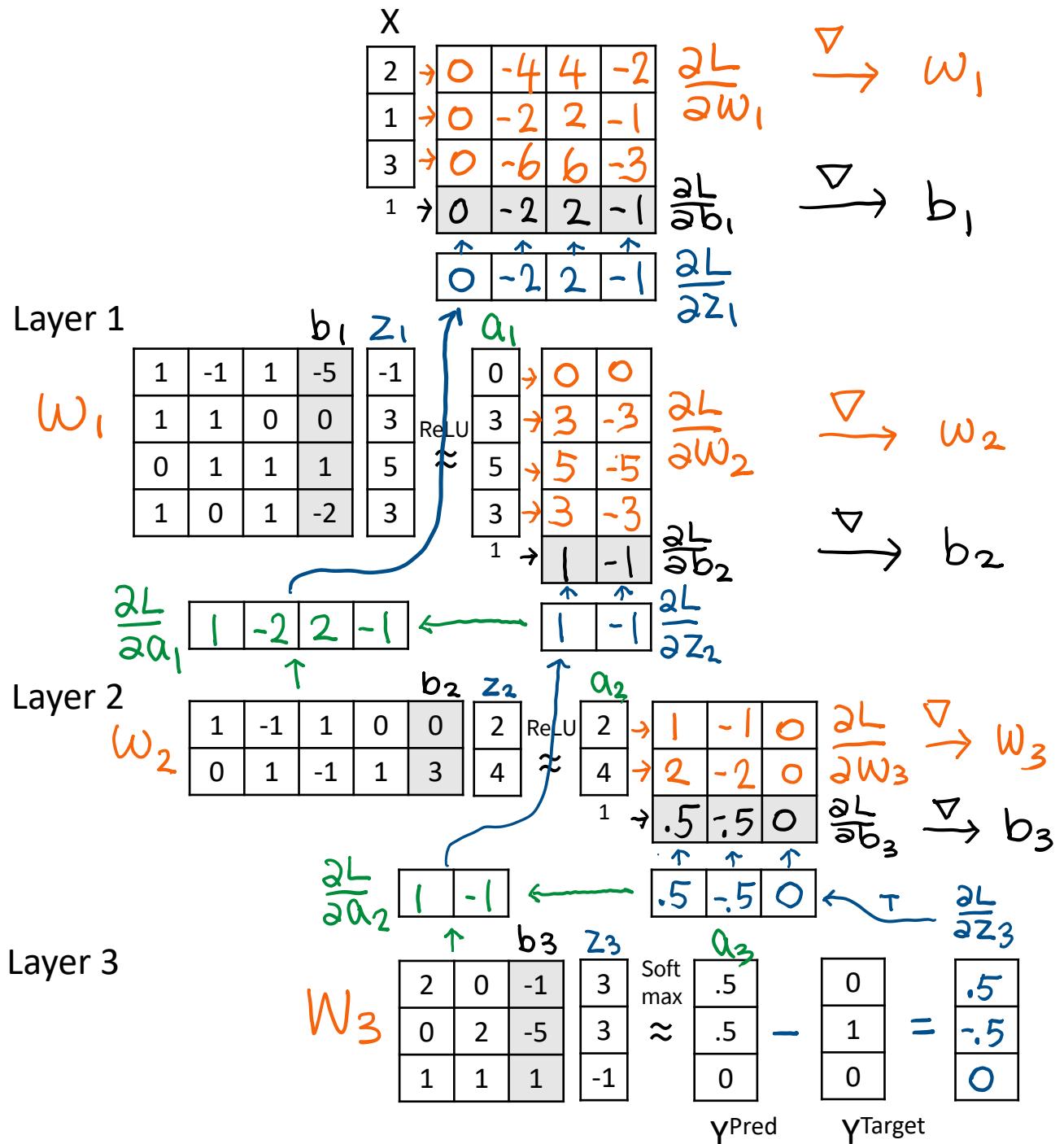
Activation Function: { ReLU | Tanh | Sigmoid }

in\_features: { int }

out\_features: { int }

bias: { T | F }

# 7. Backpropagation



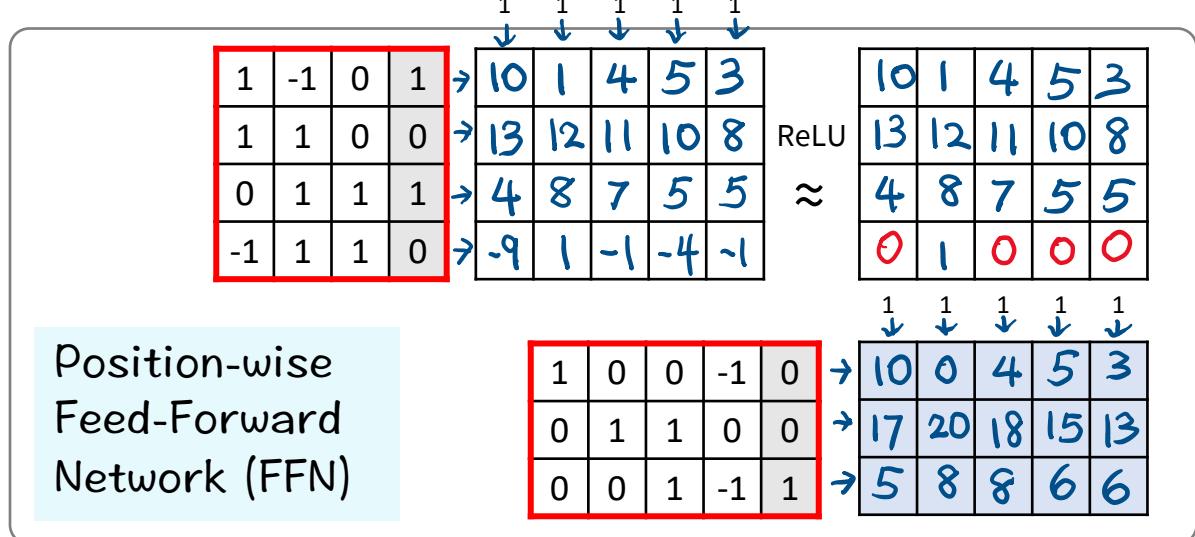
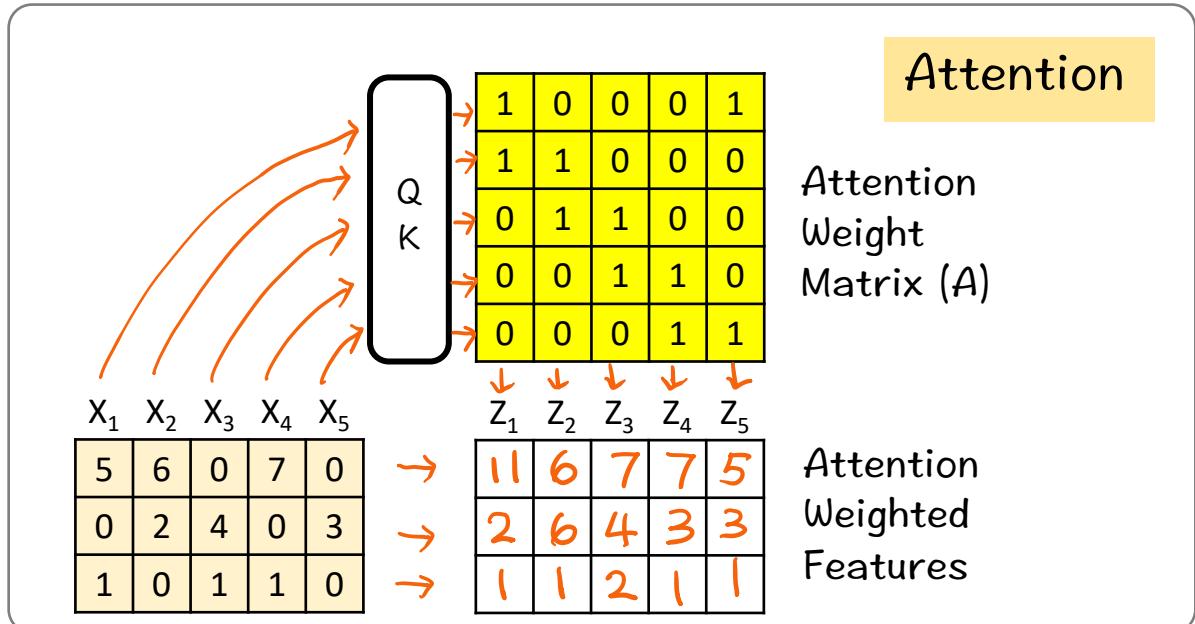
$L$ : Cross-Entropy Loss

# 8. Transformer



Features from the  
Previous Block

↓ ↓ ↓ ↓ ↓



↓ ↓ ↓ ↓ ↓

Next Block

# 9. Batch Normalization

Mini-batch:  $x_1 \ x_2 \ x_3 \ x_4$

1	0	3	0
0	3	1	1
2	1	0	2

Linear Layer

1	0	1	0	→	3	1	3	2
1	1	0	-1	→	0	2	3	0
0	2	-1	0	→	-2	5	2	0

ReLU  
≈

3	1	3	2
0	2	3	0
0	5	2	0

$\Sigma$	$\mu$	$\sigma^2$	$\sigma$
9	2	1	1
5	1	1	1
7	2	4	2

Normalize

$$\begin{array}{r} \mu \\ - \\ \hline 2 \\ 1 \\ 2 \end{array}$$

Sum ( $\Sigma$ )  
Mean ( $\mu$ )  
Variance ( $\sigma^2$ )  
Std Dev ( $\sigma$ )

1	-1	1	0
-1	1	2	-1
-2	3	0	-2

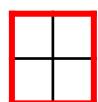
$$\begin{array}{r} \sigma \\ \div \\ \hline 1 \\ 1 \\ 2 \end{array}$$

1	-1	1	0
-1	1	2	-1
-1	1	0	-1

Scale & Shift

2	0	0	0	→	2	-2	2	0
0	3	0	0	→	-3	3	6	-3
0	0	-1	1	→	2	0	1	2

Trainable Parameters



Next Layer

# 10. Generative Adversarial Network (GAN)



Generator

Noise: N <sub>1</sub> N <sub>2</sub> N <sub>3</sub> N <sub>4</sub>			
1	1	0	1
1	0	1	-1

1   1   0	→	2   1   1   0
0   1   2	→	3   2   3   1
-1   1   0	→	0   1   1   0

[≈ ReLU]

Fake: F <sub>1</sub>	1   1   2   1
F <sub>2</sub>	2   1   2   0
F <sub>3</sub>	3   2   4   1
F <sub>4</sub>	1   1   2   1

[≈ ReLU]

Real:

X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>
2	3	3	4
1	1	1	1
2	3	4	3
1	1	1	1

1   2   2   3
3   4   5   4
2   3   4   3
1   2   2   3

[≈ σ]

Predictions:

Y	.7	.5	.9	.3
---	----	----	----	----

.7	.9	.9	1
----	----	----	---

Training the Discriminator

Targets:

-	Y <sub>D</sub>	0   0   0   0
---	----------------	---------------

1   1   1   1
---------------

Loss Gradients:

$\frac{\partial L_D}{\partial Z}$	.7	.5	.9	.3
-----------------------------------	----	----	----	----

-.3	-.1	-.1	0
-----	-----	-----	---

Training the Generator

Targets:

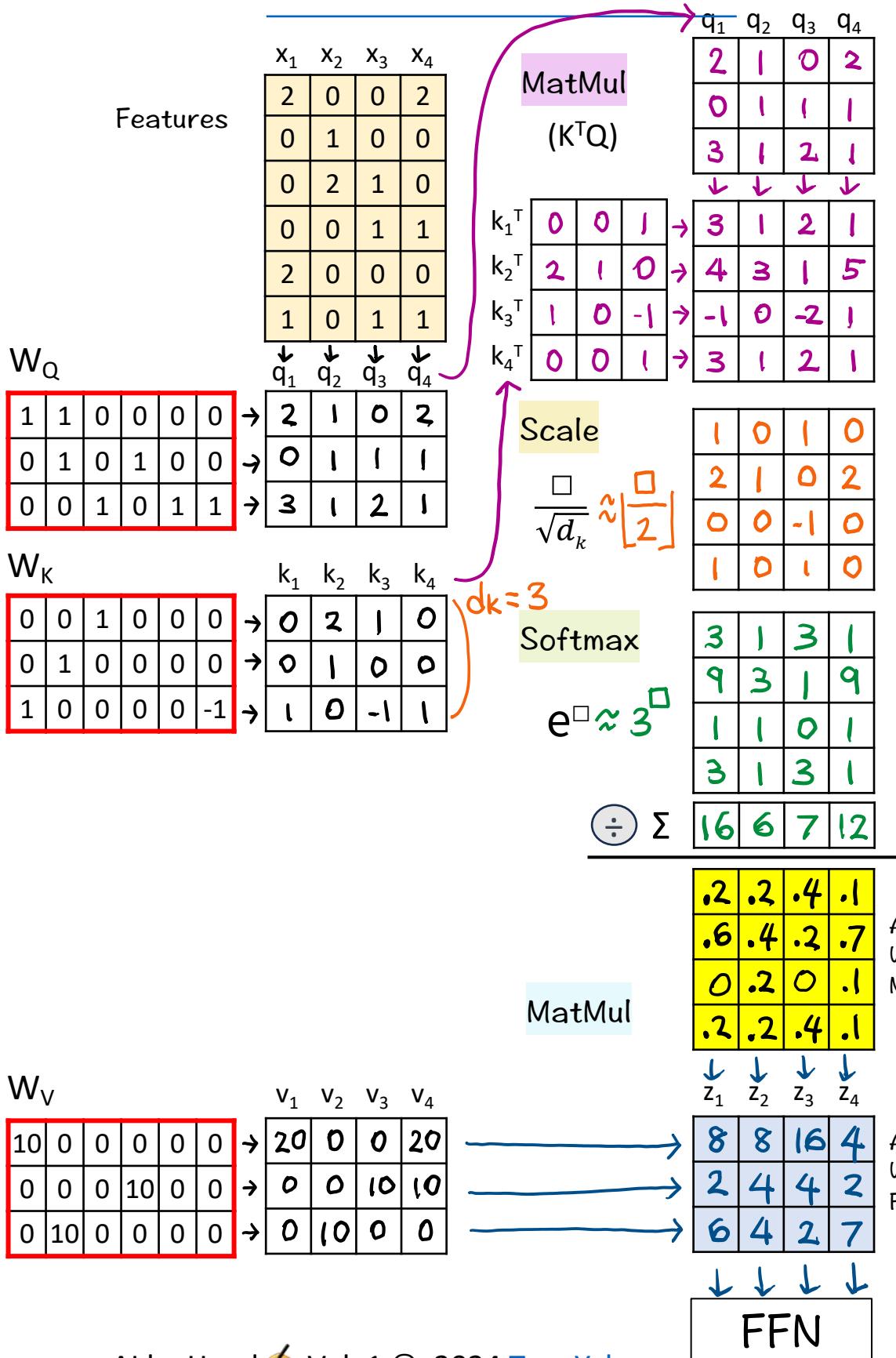
-	Y <sub>G</sub>	1   1   1   1
---	----------------	---------------

Loss Gradients:

$\frac{\partial L_G}{\partial Z}$	-.3	-.5	-.1	-.7
-----------------------------------	-----	-----	-----	-----

# 11. Self Attention

1376



# 12. Dropout



557

Random Sequence

.61
.39
.75
.40
.65
.42
.23

Linear

Training Data:

3	5
4	1
1	1

1	0	0
1	1	0
0	1	1
1	-1	0

3	5
7	6
5	2
4	X

[≈ ReLU]

Inference

Unseen Data:

3	3
2	1
1	1

1	0	0
1	1	1
-1	1	1
1	-1	0

3	3
6	5
0	X
1	2

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

3	3
6	5
0	0
1	2

Dropout  
( $p=0.5$ )

2	0	0	0
0	0	0	0
0	0	2	0
0	0	0	0

6	10
0	0
10	4
0	0

Linear

1	0	0	1	0
0	1	1	0	0
1	0	-1	-1	1

6	10
10	4
6	X

[≈ ReLU]

Dropout  
( $p=0.33$ )

1.5	0	0
0	1.5	0
0	0	0

9	15
15	6
0	0

Linear

1	-1	0	0
0	1	-1	-2

-6	9
13	4

Outputs  
Y

1	1	0	0
0	1	-1	-1

13	13
4	3

Training



-4	7
10	5

-2	2
3	-1

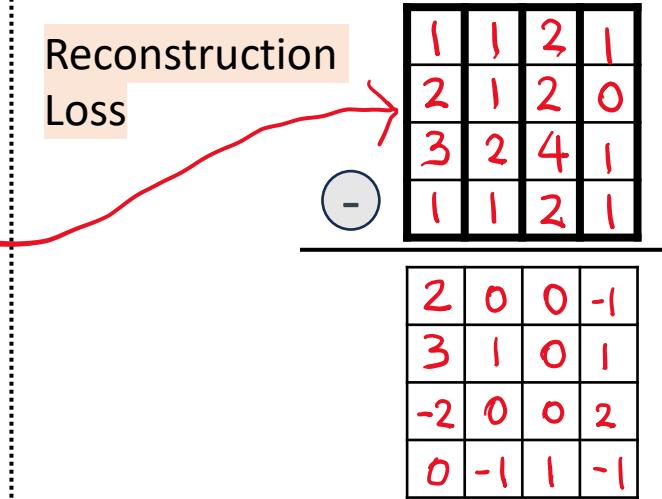
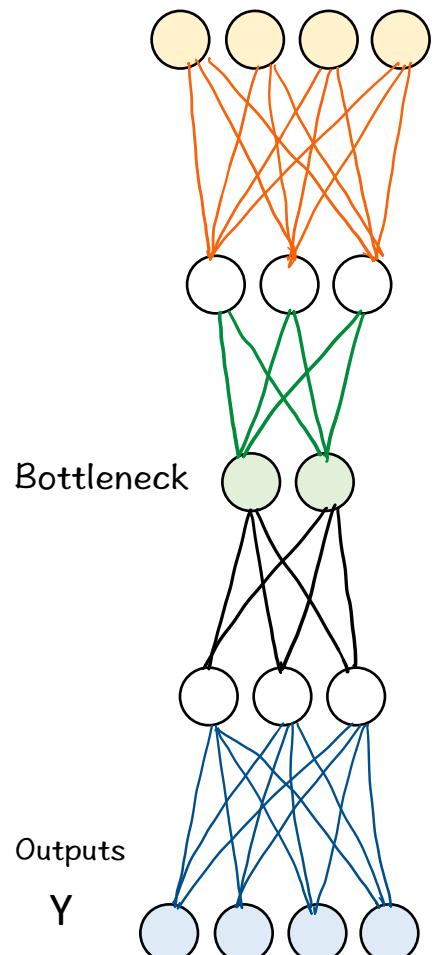
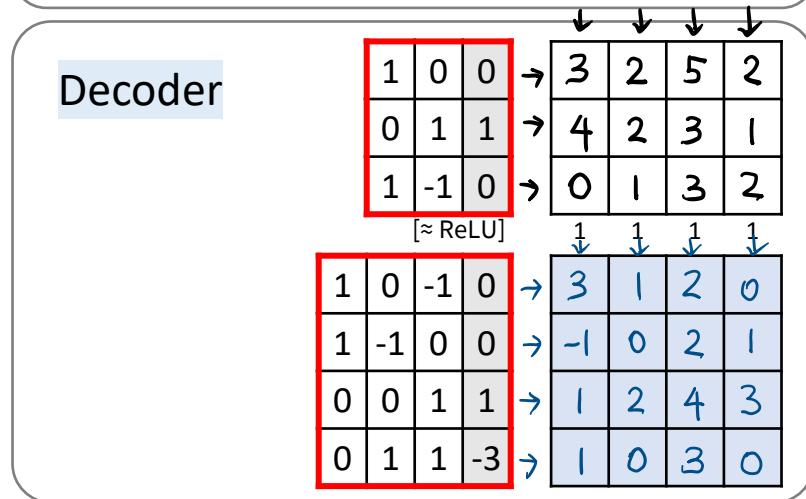
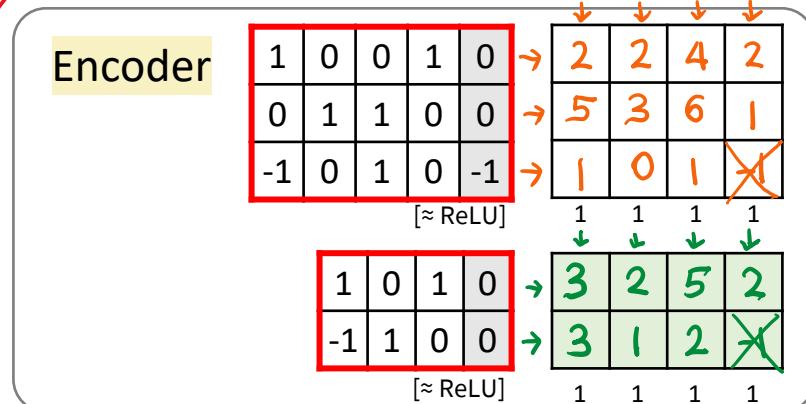
Targets  
Y'

-4	4
6	-2

MSE Loss  
Gradients  
 $\frac{\partial L}{\partial Y}$

# 13. Autoencoder

 849



# 14. Vector Database

2224

Query

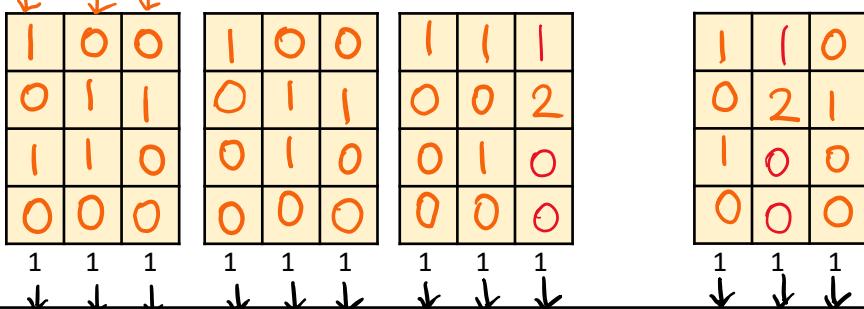
Data [how are you] [who are you] [who am I]

[am I you]

Word Embeddings

a	an	the	how	why	who	what	are	is	am	be	was	you	we	I	they	she	he	she	me	him	her
0	-1	0	1	0	1	0	0	-1	1	0	0	0	3	1	0	-1	0	0	0	-1	0
2	0	2	0	0	0	-1	1	0	0	0	2	1	0	2	0	2	0	0	2	0	0
-1	0	-1	1	2	0	0	1	0	1	-1	0	0	-1	0	3	0	0	-1	0	2	-1
0	1	0	0	1	0	1	0	1	0	1	-2	0	0	0	1	0	1	0	1	0	1

Text Embeddings



Encoder

Linear & ReLU

1	1	0	0	0
0	1	0	1	0
1	0	1	0	-1
1	-1	0	0	0

1	1	1
0	1	1
0	0	2
0	1	0
0	1	3

1	3	1
0	2	1
1	0	X
1	X	X
1	X	X

Mean Pooling

$$\Sigma / 3$$

3/3
2/3
1/3
1/3

3/3
2/3
0

5/3
2/3
1/3
1/3

5/3
3/3
1/3
1/3

Indexing

Projection

1	1	0	0
0	0	1	1

5/3
2/3

5/3
0

7/3
2/3
0

Vector Storage

8/3
2/3

Retrieval

Dot Products

44/9

40/9

60/9

$\begin{bmatrix} 8/3 \\ 2/3 \end{bmatrix}$

T

Nearest Neighbor (argmax)

✓

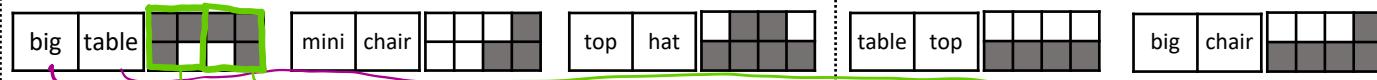
who am I

# 15. CLIP

885

400 millions more ...

mini batch of text-image pairs



word2vec

	mini	big	top	hat	chair	table
0	1	1	0	1	0	0
1	0	0	0	1	1	1
2	1	0	1	0	1	1

Word  
Embeddings

1	0	0	1	0
0	1	1	1	0
1	1	0	0	1

Text Encoder

1	0	1	0	→	2	1	0	1	1
0	3	0	-2	→	2	1	1	1	-2
1	1	0	1	→	2	2	2	3	2

[Mean Pooling]  
(round)

2	1	1
1	1	3
2	0	2

[Projection]

1	1	0	-2
0	1	1	-2

1	0	-1
1	2	0

T<sub>1</sub> T<sub>2</sub> T<sub>3</sub>

1	0	-1
1	2	0

[Softmax]

$$e^{\square} \approx 3^{\square}$$

$$\frac{\Sigma}{\Sigma}$$

Shared Embedding Space

Similarity  
Image → Text

Target

Cross Entropy  
Loss Gradients

Image → Text

-1	.1	0
.75	.25	0
.7	.3	-1

Text → Image

-8	.1	0
.1	-.9	0
.7	.8	-1

2	0
1	0
2	1

2	0	-2
3	1	0
27	9	0

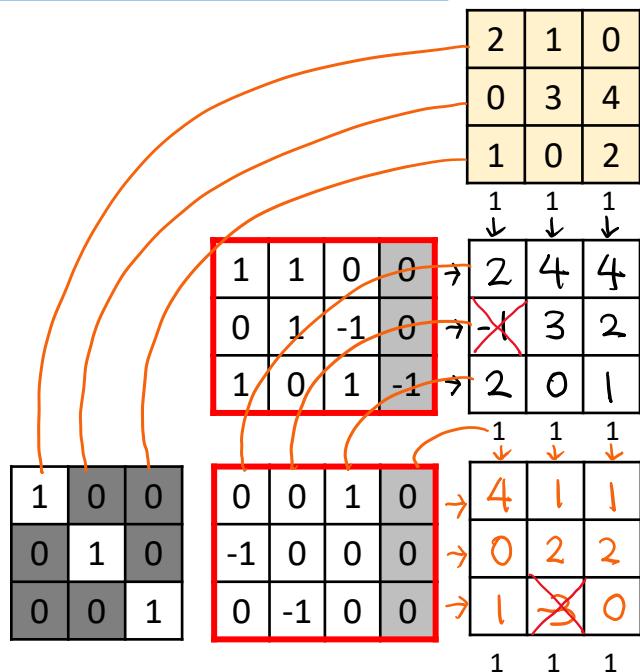
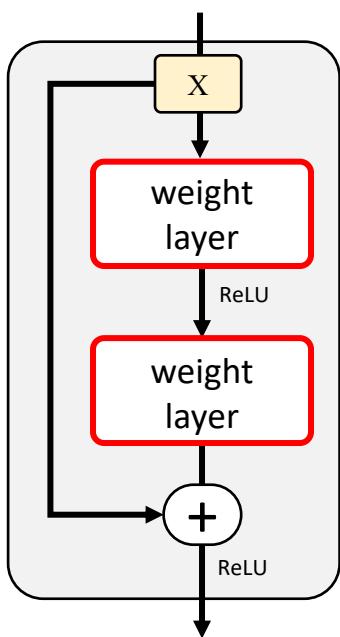
.9	.1	0
.75	.25	0
.7	.3	0

1	0	0
0	1	0
0	0	1

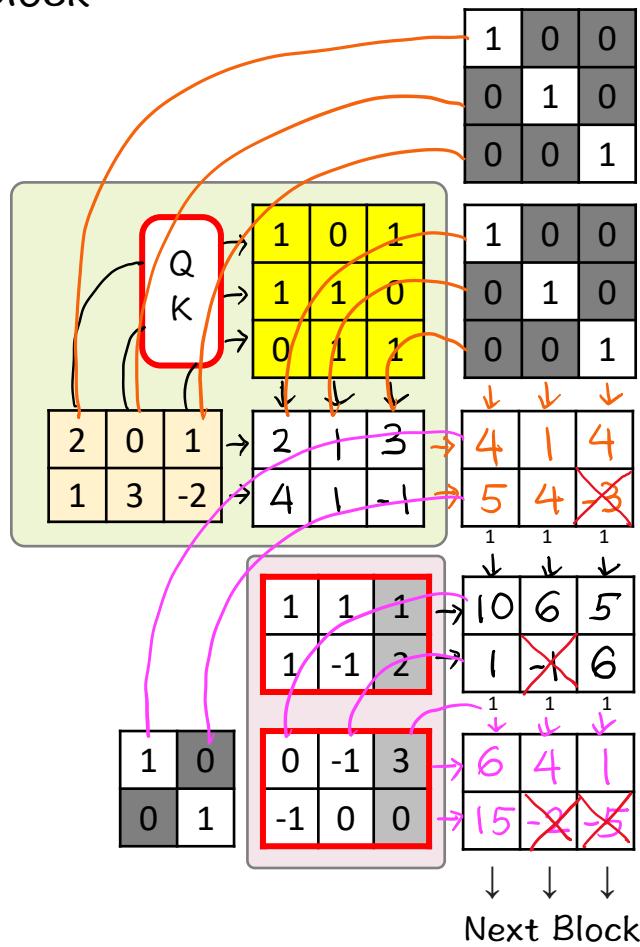
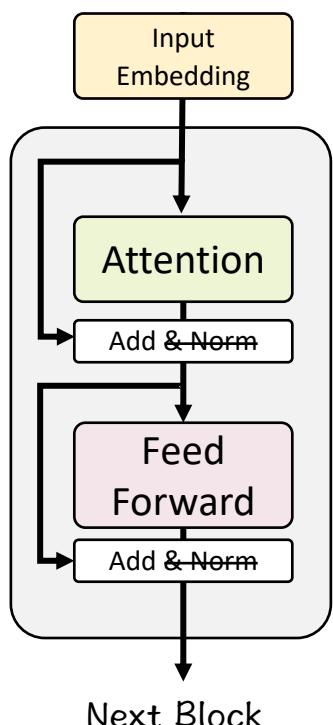
Similarity  
Text → Image

.2	.1	0
.1	.1	0
.7	.8	0

# 16. Residual Network

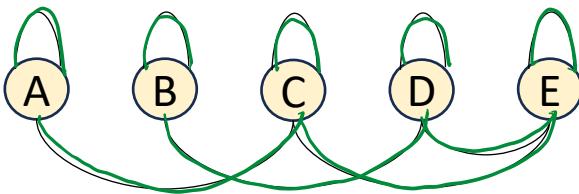


Transformer's Encoder Block



# 17. Graph Convolutional Network

Graph Data



573

Graph  
Convolutional  
Network

A	B	C	D	E
2	0	1	0	1
1	1	0	0	0
0	0	-1	1	1
0	3	0	1	0

1	1	0	0	0
0	1	0	-1	0
1	0	0	1	-1

[ReLU]

1	1	0	0	0
0	1	1	0	1
1	2	0	0	0

Messages

A	B	C	D	E
1		1		
	1		1	
1		1	1	1
			1	1

1	1	1	0
0	1	3	1
1	2	3	0

Messages

A	B	C	D	E
1		1		
	1		1	
1		1	1	1
			1	1

↓ ↓ ↓ ↓ ↓

1	0	0	-2
0	1	0	-2
0	0	1	-5
1	-1	0	0
1	0	-1	0

[ReLU]

1	1	1	1	1
-4	4	5	3	0
0	1	1	1	0.5

$\sigma$

# 18. SORA's Diffusion Transformer

2118

Training Video

1	0	2	0	0	1	0	1
0	1	1	0	3	0	4	0
1	0	0	1	1	3	0	0
0	1	4	0	0	1	4	0

Spacetime  
Patches  
(Pixels)

1	0	0	1
2	0	1	0
0	1	3	0
0	1	4	0

Diffusion

Step  $t = 3$

1
1

Prompt  
“sora is sky”

Text  
Encoder

0
1
-1

0
1
-1

Visual Encoder

1	0	-1	0	0
0	1	0	1	1

1	0	0	1	0
0	1	0	-1	1
1	1	0	0	1
0	2	0	1	0

[ReLU]

Sampled  
Noise



0	2	1	-1
-1	0	-2	1

Noised  
Latent

1	2	1	0
2	2	4	2

2	5	0	X
1	3	5	4
2	7	4	2
0	X	6	6

Predicted  
Noise



1	4	-1	-2
1	3	5	4

Visual Decoder

1	0	1
0	1	0
1	1	0
-1	1	0

2	5	0	X
1	3	5	4
2	7	4	2
0	X	6	6

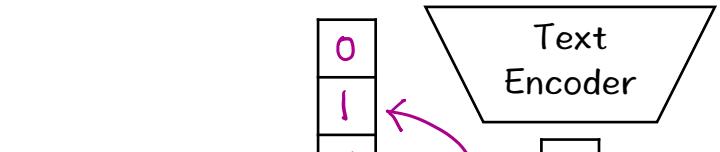
[ReLU]

Diffusion

Step  $t = 3$

1
1

“sora is sky”



1	0	0	0
0	1	0	-1
1	1	0	0
0	2	0	1

2
-1
-1
5

Adaptive  
Layer Norm



Q	K
1	1
0	1
0	1

1	3	1	-1
3	3	1	3

4	4	0	-1
6	4	4	3

-1	1	-2
0	1	-5

0	-2	2	2
1	-1	-1	-2

Train

Sampled  
Noise



0	2	1	-1
-1	0	-2	1

MSE Loss  
Gradients

0	-4	1	3
2	0	1	-3

Generated Video

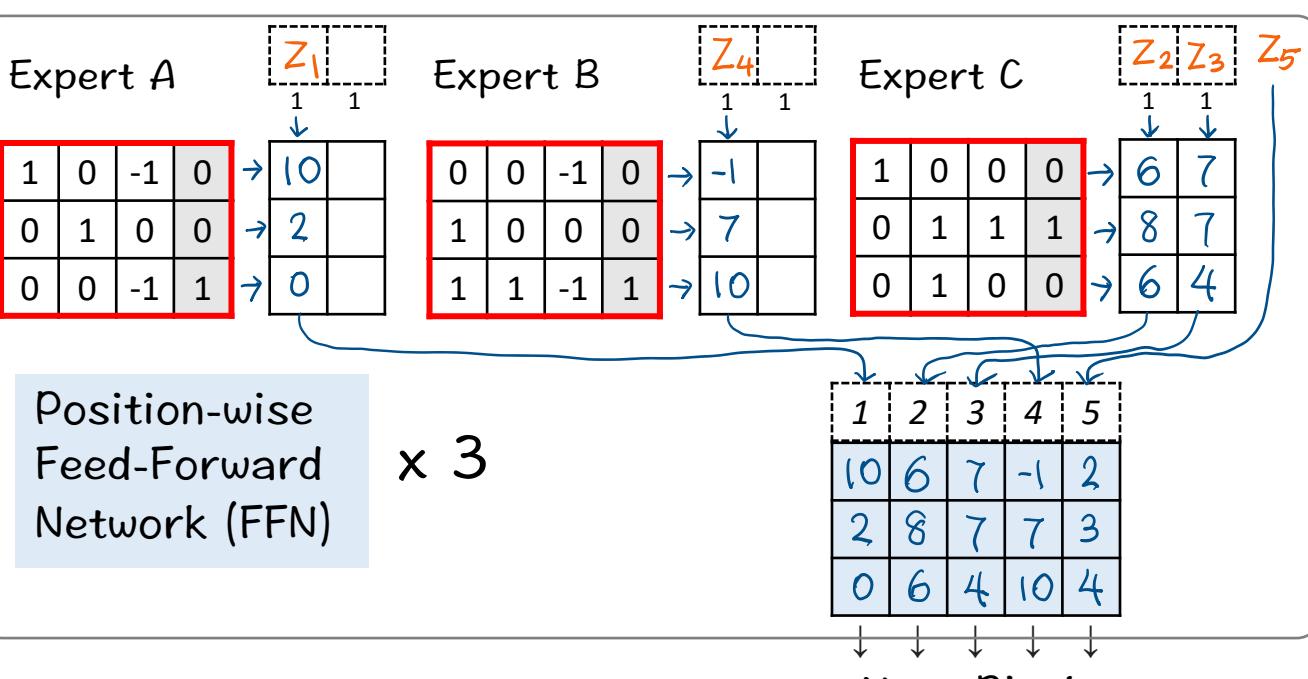
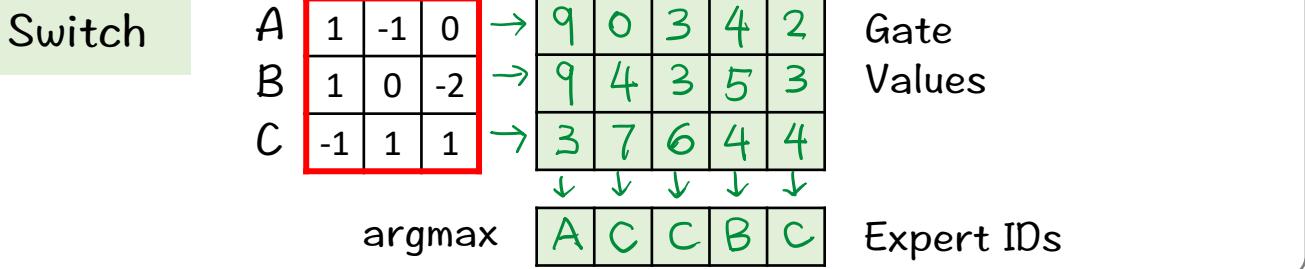
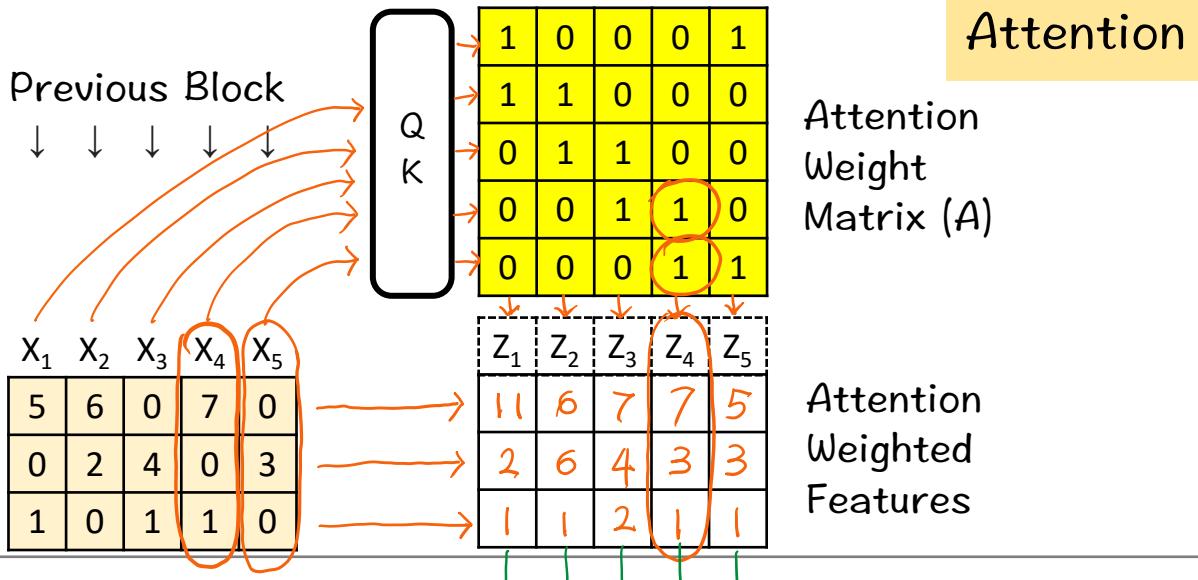
2	5
1	3
2	7
0	0

# 19. Switch Transformer

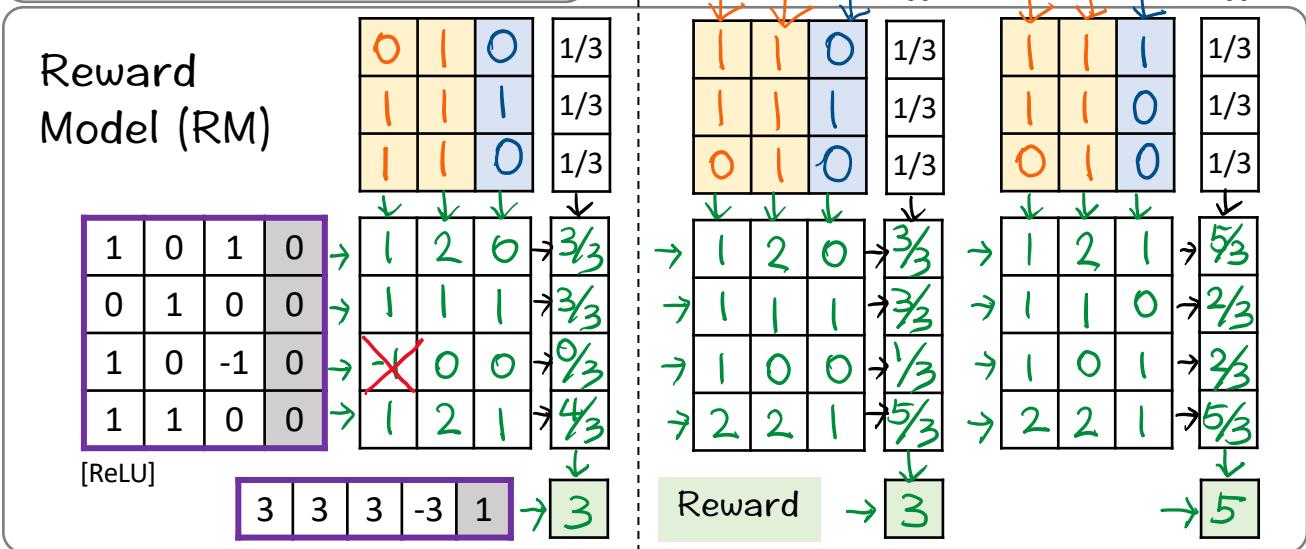
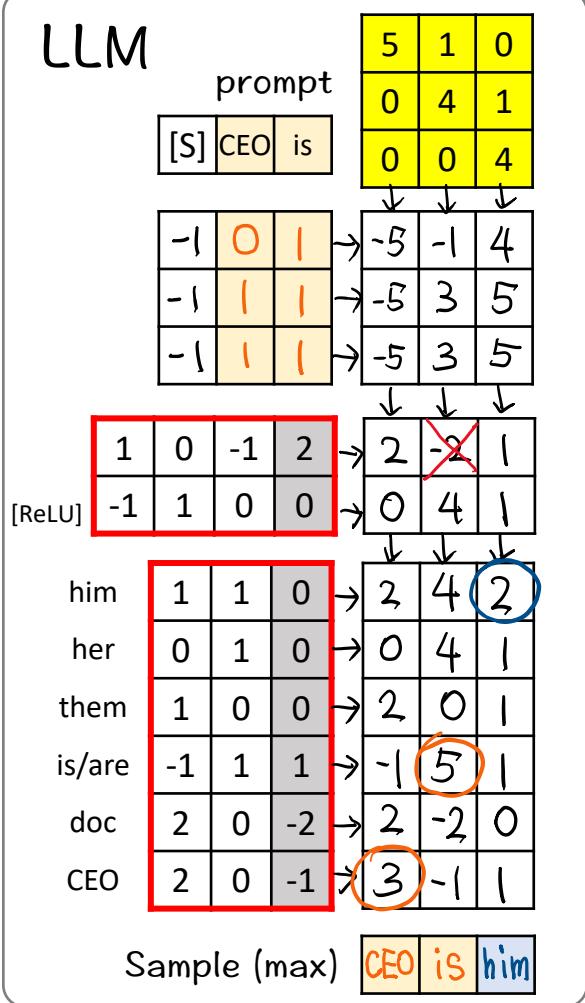


576

(Gemini 1.5's Sparse Mixture of Experts)



# 20. Reinforcement Learning with Human Feedback (RLHF)



Align LLM

Loss: -3

Loss Gradient: -1

Train RM

5 Winner - Loser: 2

3 Predicted σ: 0.9

- Target: -1

Loss Gradient: -0.1