AI by Hand ✍

Volume 1

Basic x 5
Advanced x 20

Prof. Tom Yeh

2024

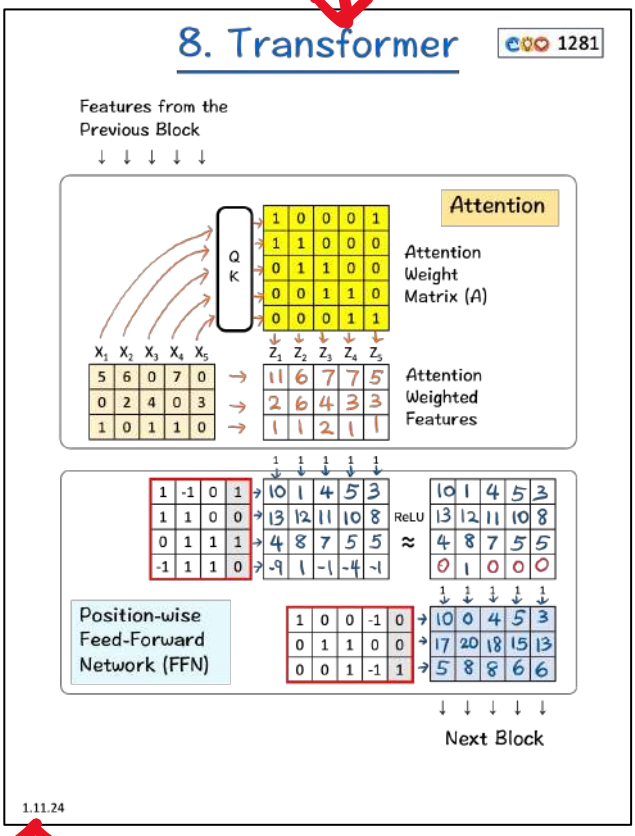
Basic

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)

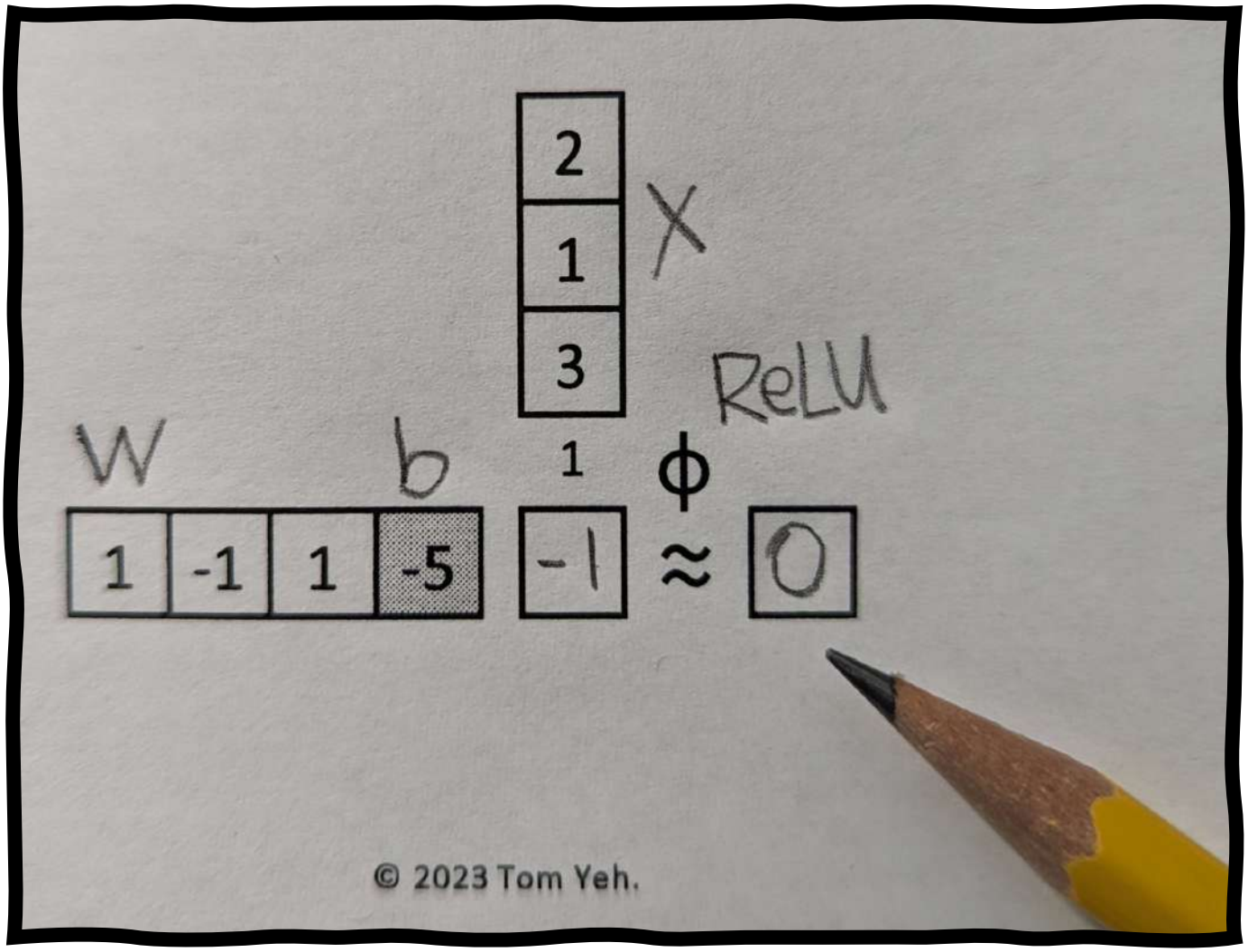
Link to my original LinkedIn post with animation and explanation



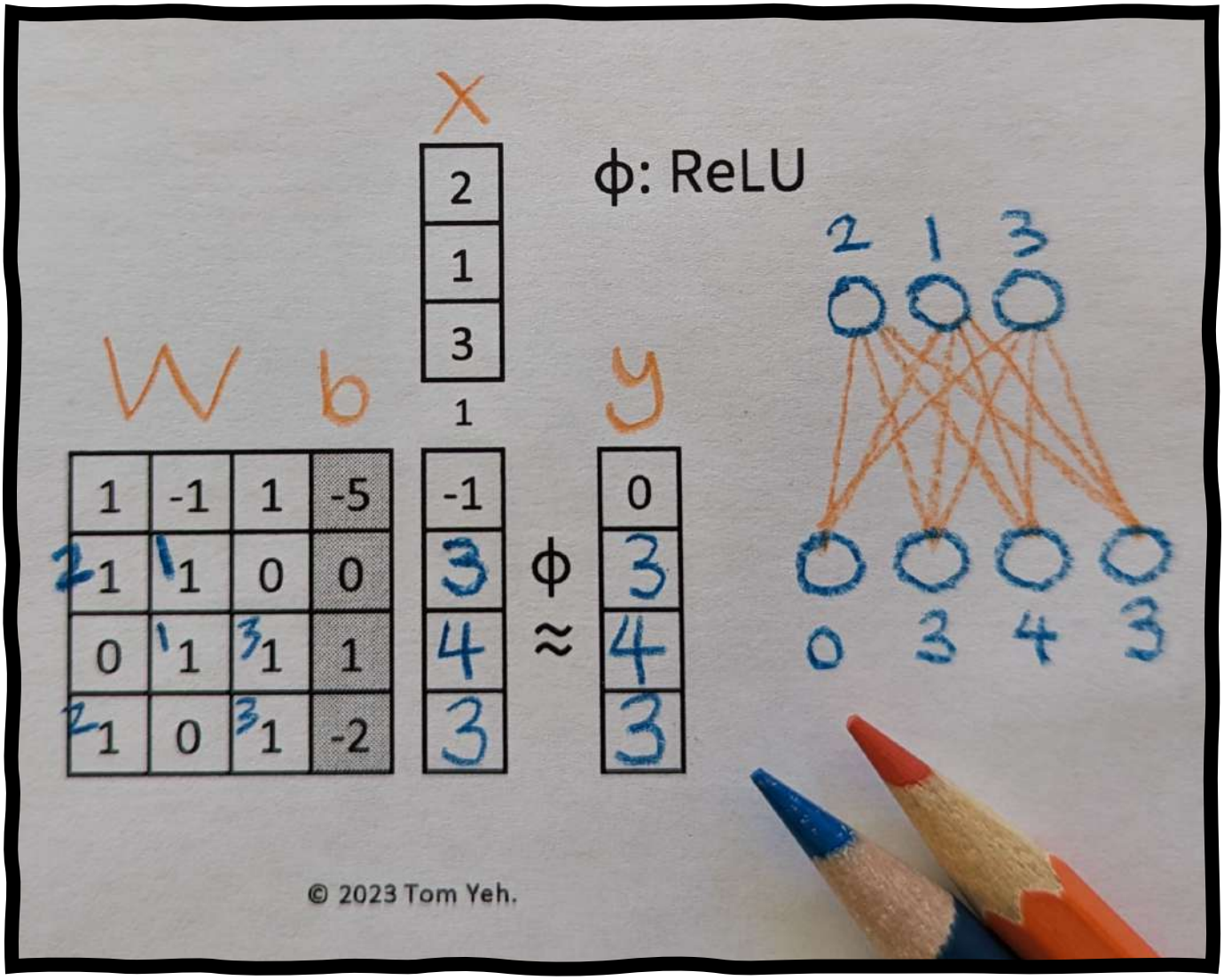
Date originally posted

I. One Node

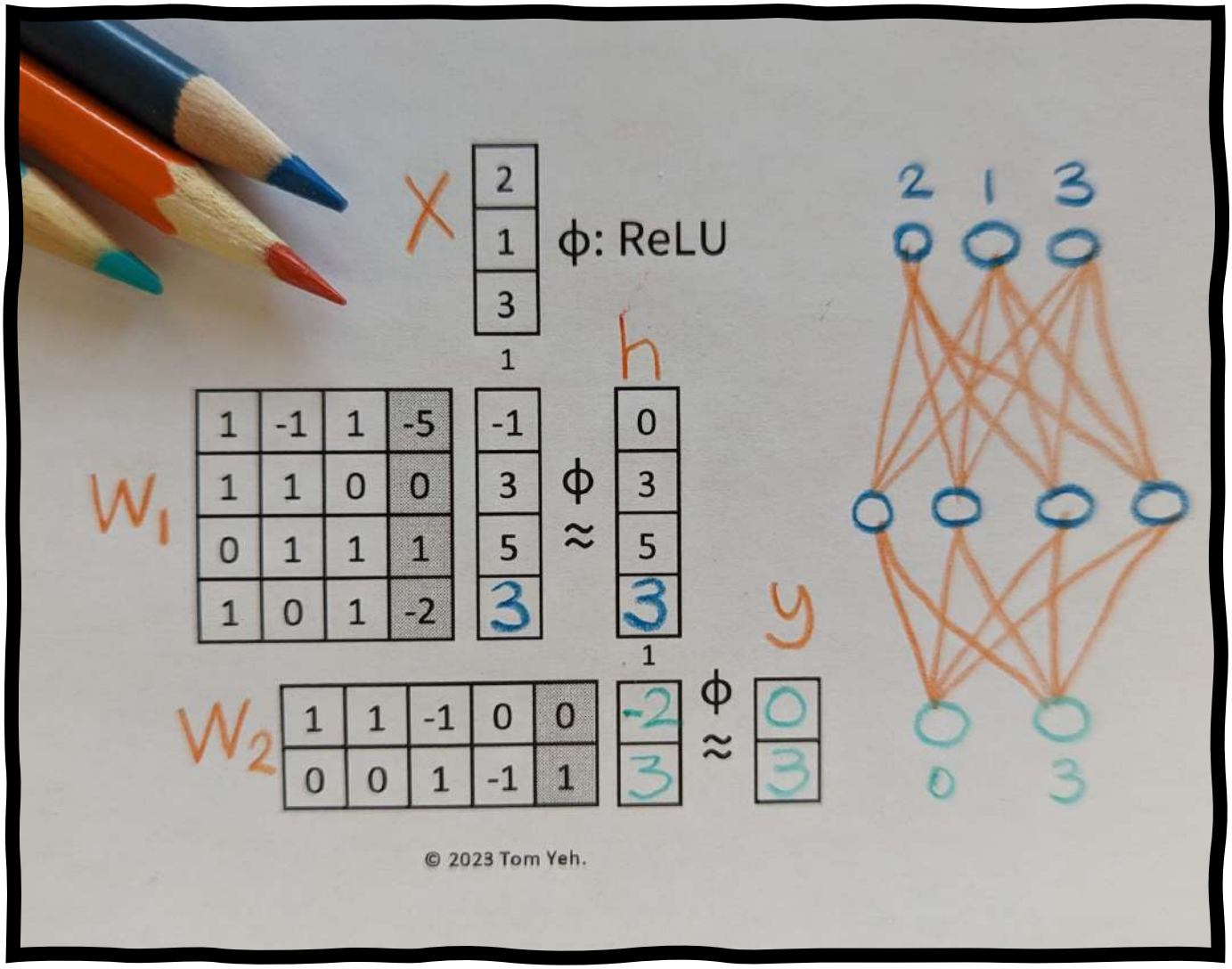
406



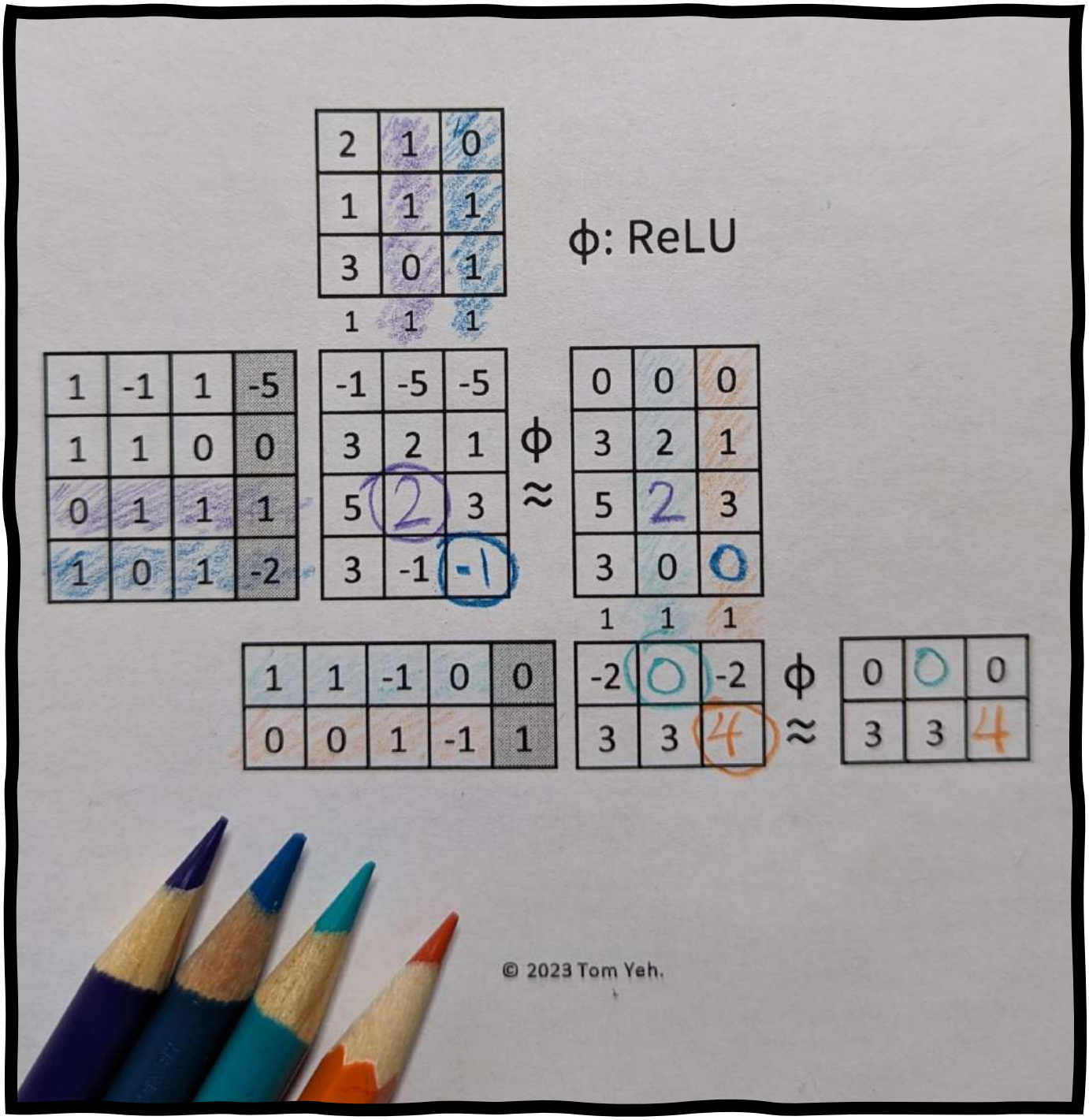
II. Four Nodes



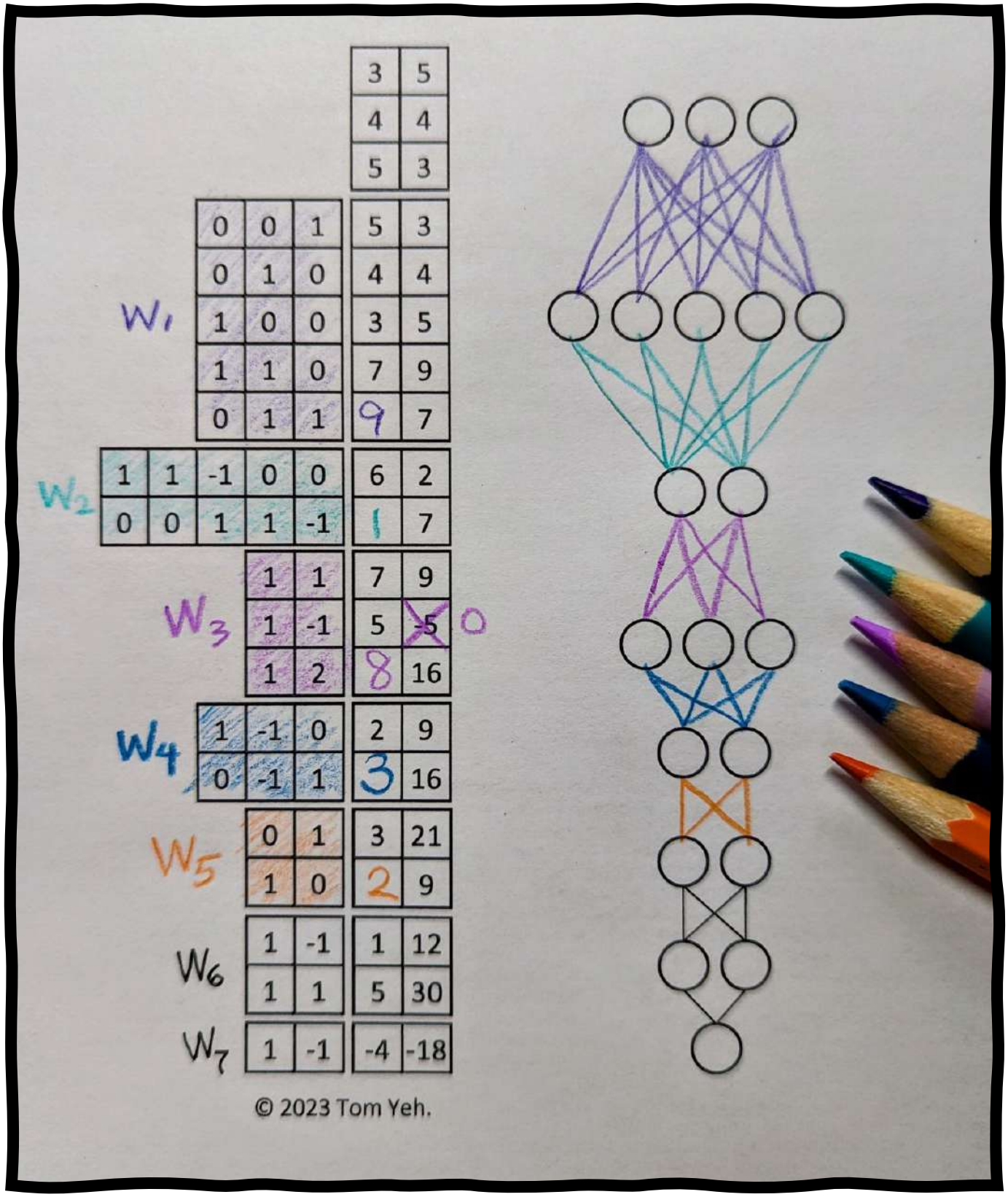
III. Hidden Layer



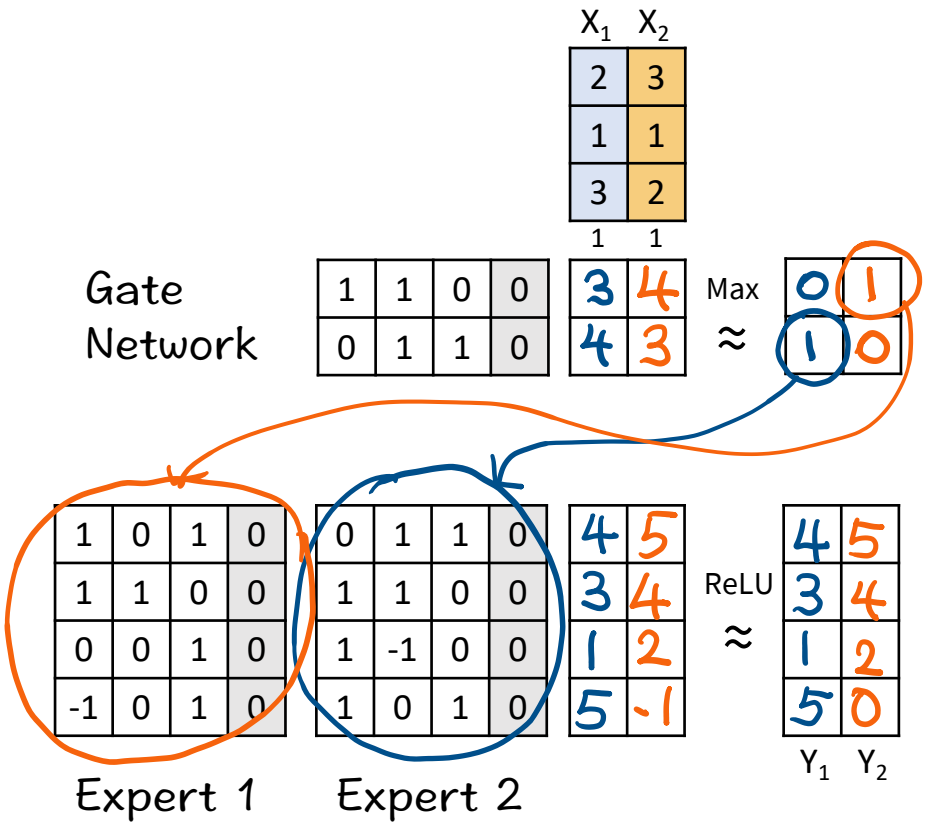
IV. Three Inputs



V. Seven Layers



1. Mixture of Experts



2. Recurrent Neural Network (RNN)

406

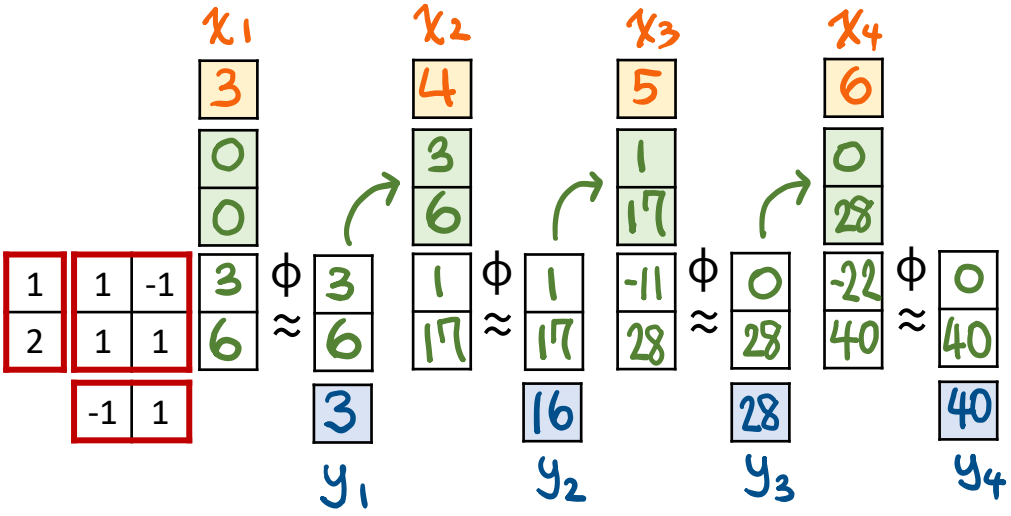
Input Sequence X | 3 | 4 | 5 | 6 |

Parameters A | 1 | -1 | / | 1 | 1 | B | 1 | / | 2 | C | -1 | 1 |

Activation Function ϕ: ReLU

Hidden States H_0 | 0 | / | 0 |

Output Sequence Y | | | | |



3. Mamba's S6 Model

Input Sequence 3 4 5 6

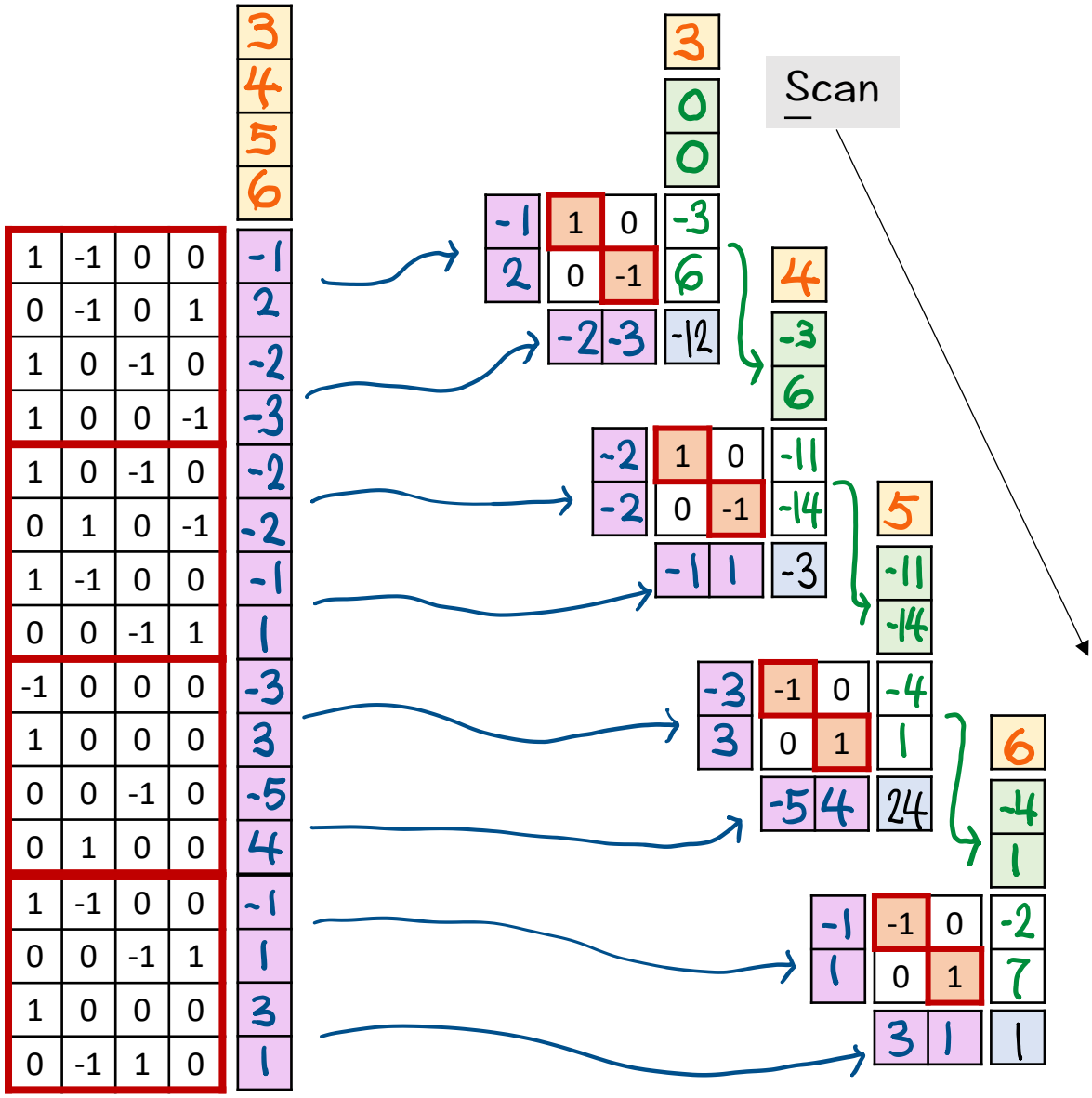
Output Sequence -12 -3 24 1

Parameters

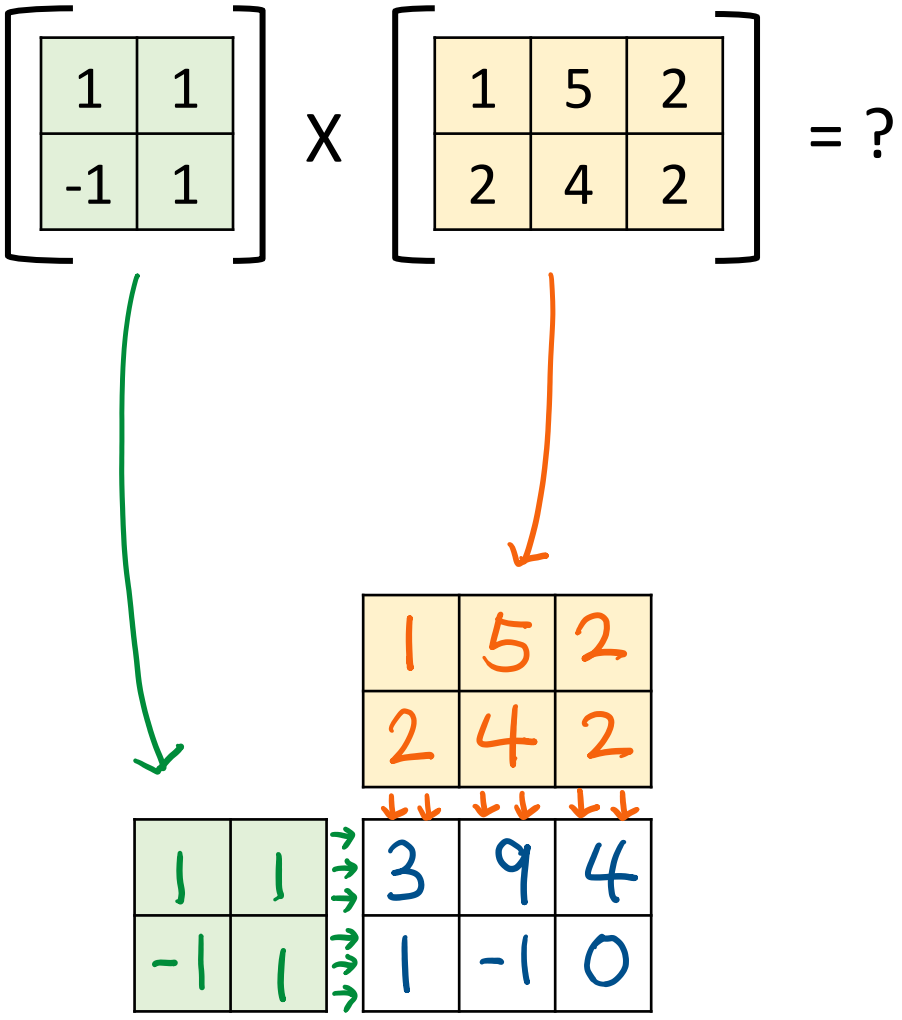
Selective

Structured

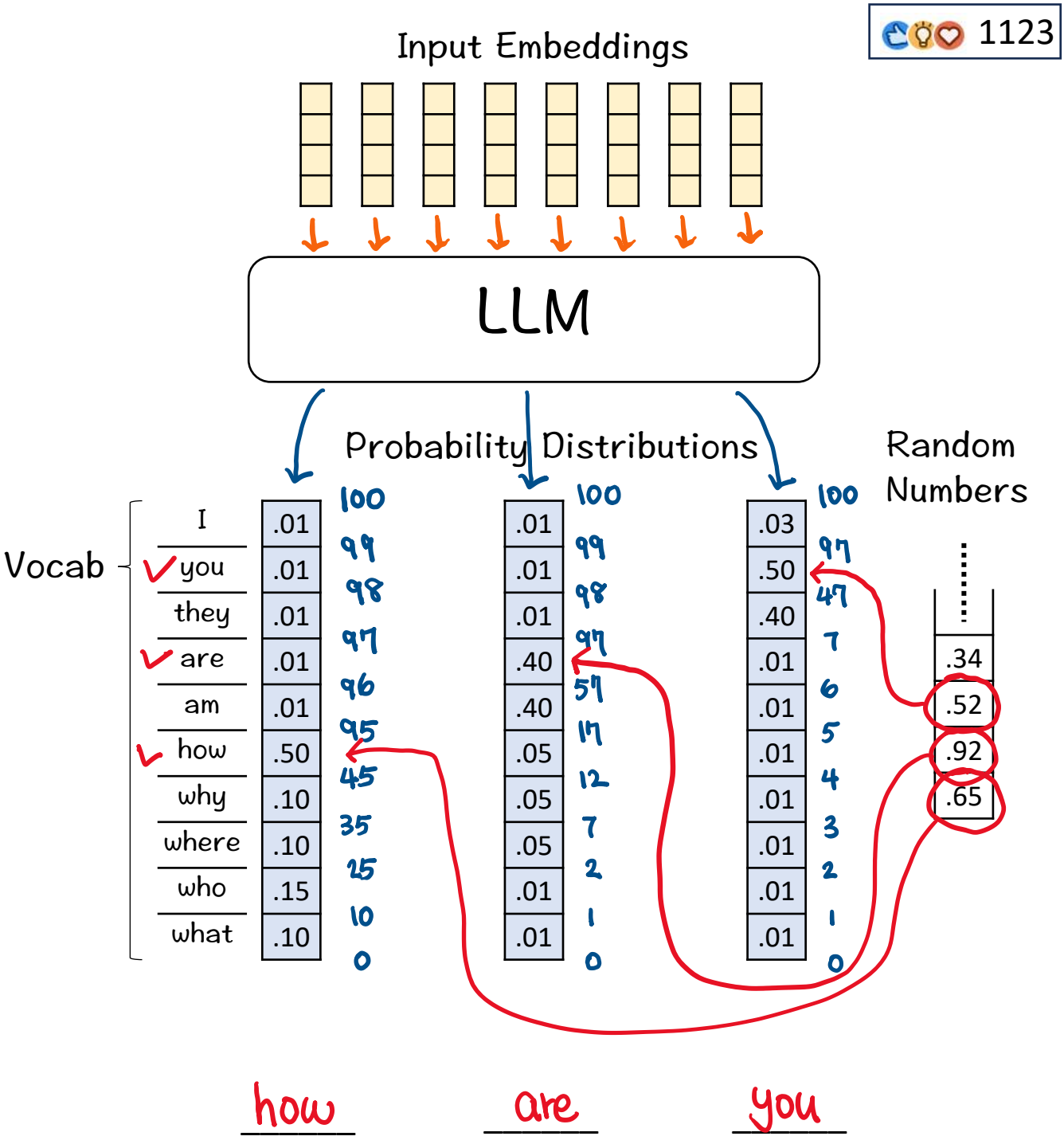
State-Space



4. Matrix Multiplication



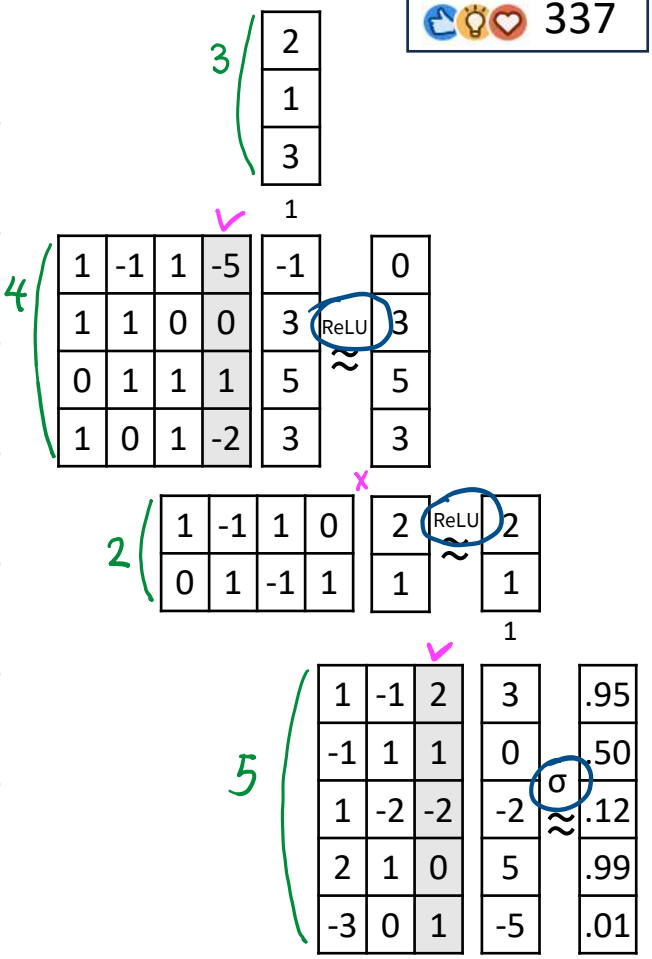
5. How does an LLM sample a sentence?



6. Multi Layer Perceptron in pytorch

337

```
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 }

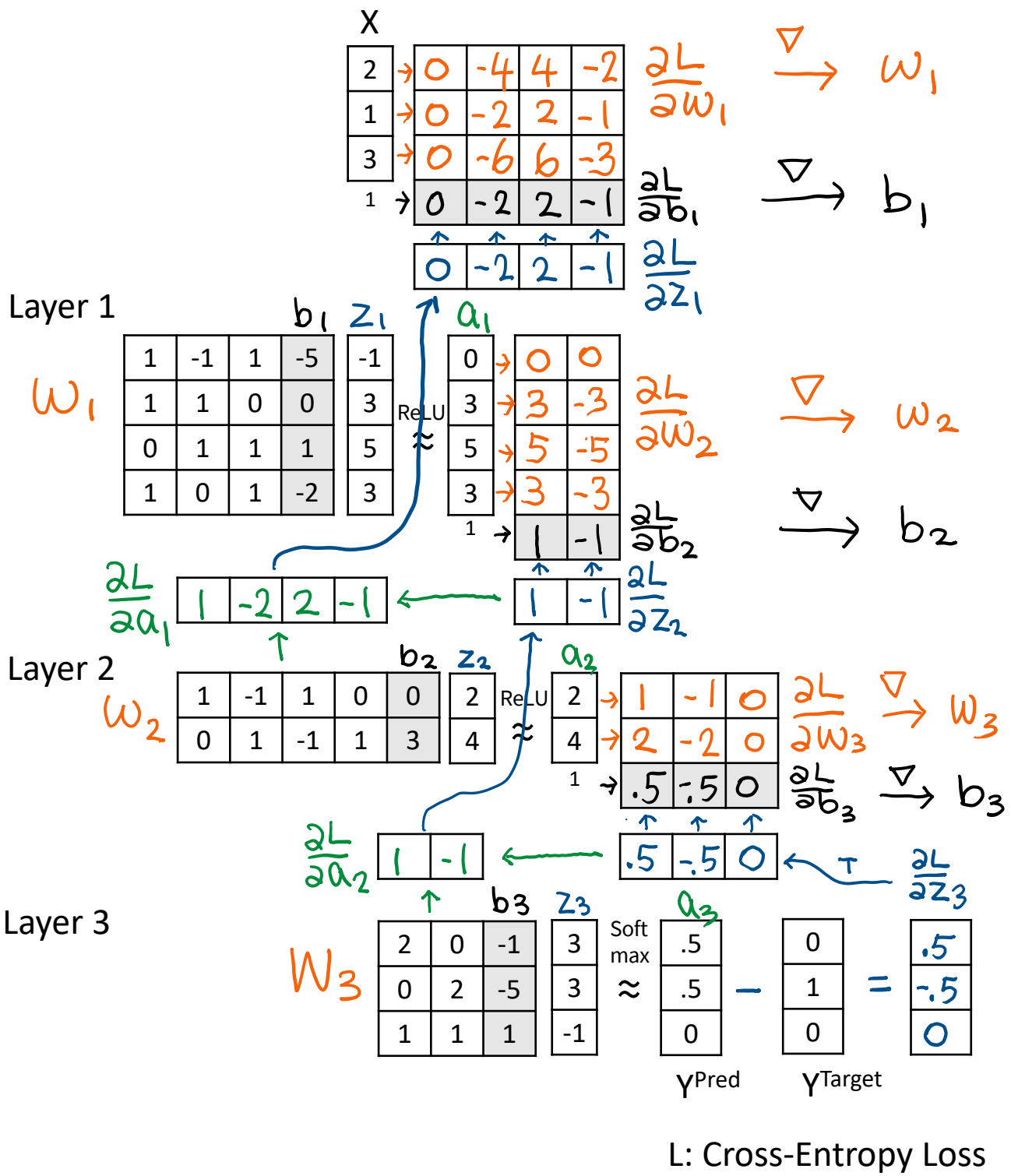
in_features: { int }

out_features: { int }

bias: { T | F }

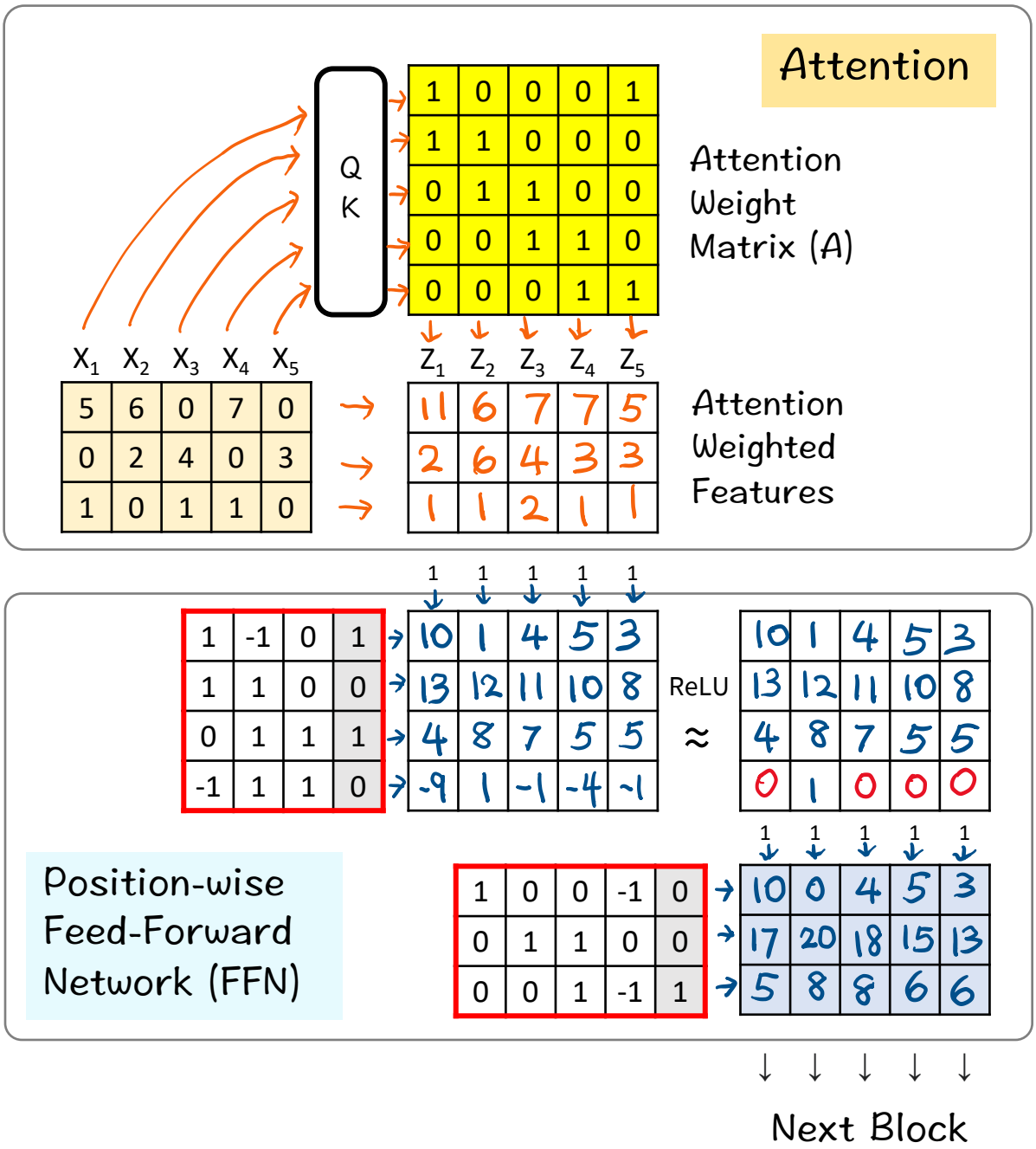
7. Backpropagation

1260



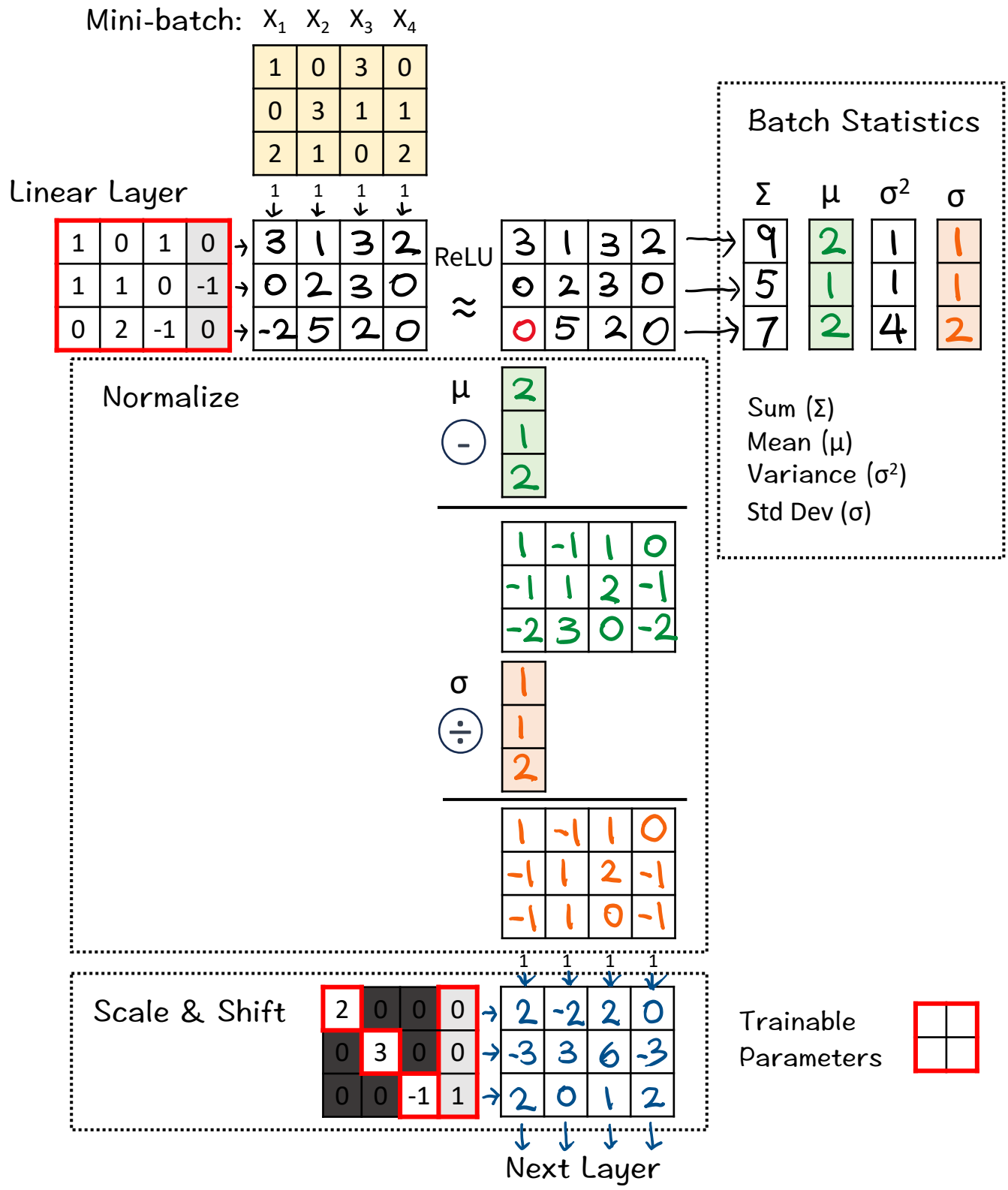
8. Transformer

Features from the Previous Block

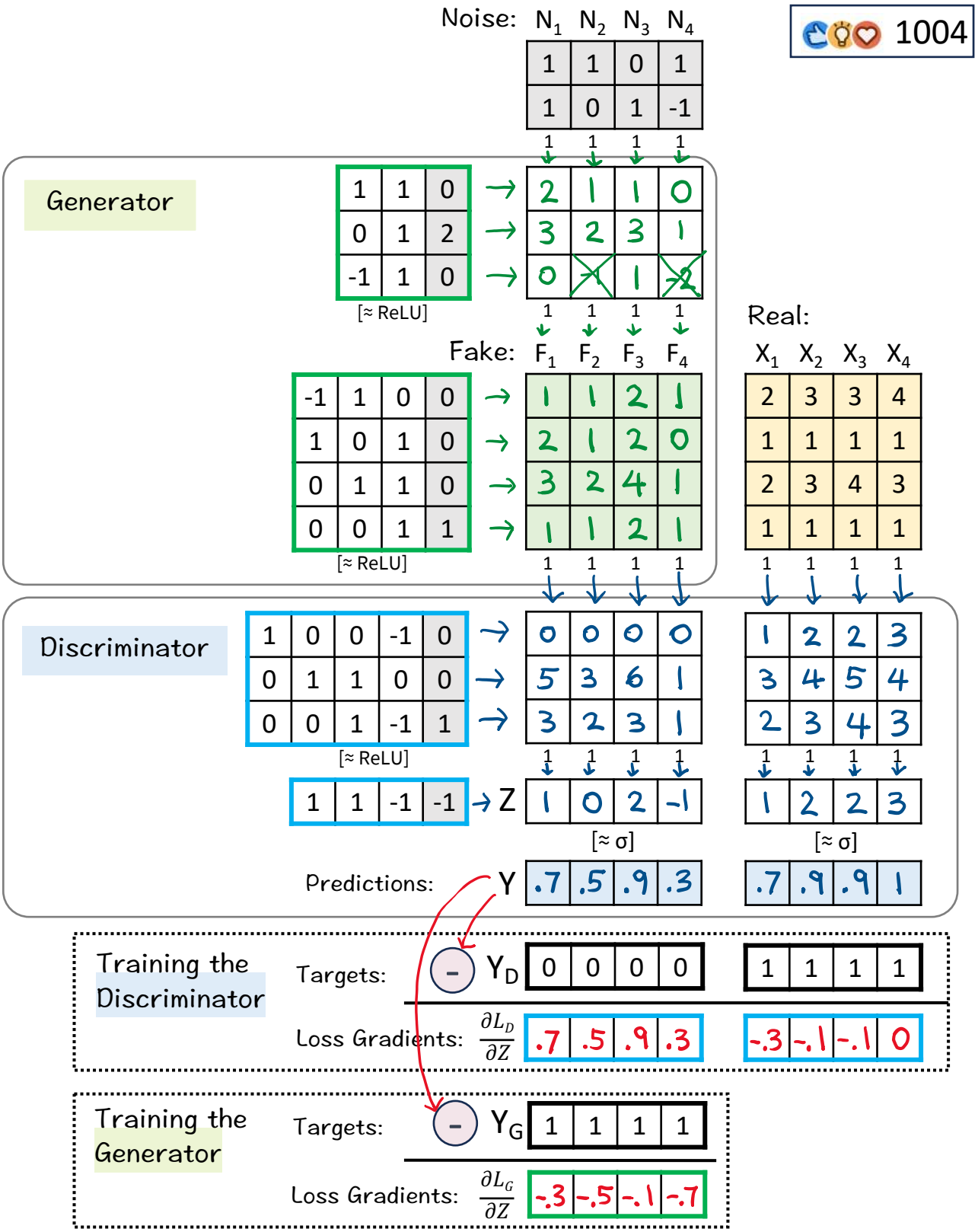


9. Batch Normalization

504



10. Generative Adversarial Network (GAN)



11. Self Attention

Features

	x_1	x_2	x_3	x_4
	2	0	0	2
	0	1	0	0
	0	2	1	0
	0	0	1	1
	2	0	0	0
	1	0	1	1

W_Q

							q_1	q_2	q_3	q_4
1	1	0	0	0	0	→	2	1	0	2
0	1	0	1	0	0	→	0	1	1	1
0	0	1	0	1	1	→	3	1	2	1

W_K

							k_1	k_2	k_3	k_4
0	0	1	0	0	0	→	0	2	1	0
0	1	0	0	0	0	→	0	1	0	0
1	0	0	0	0	-1	→	1	0	-1	1

W_V

							v_1	v_2	v_3	v_4
10	0	0	0	0	0	→	20	0	0	20
0	0	0	10	0	0	→	0	0	10	10
0	10	0	0	0	0	→	0	10	0	0

MatMul

(K^TQ)

	q_1	q_2	q_3	q_4
	2	1	0	2
	0	1	1	1
	3	1	2	1

k_1^T	0	0	1	→	3	1	2	1
k_2^T	2	1	0	→	4	3	1	5
k_3^T	1	0	-1	→	-1	0	-2	1
k_4^T	0	0	1	→	3	1	2	1

Scale

$\frac{\square}{\sqrt{d_k}} \approx \frac{\square}{2}$

$d_k = 3$

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

Softmax

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

3	1	3	1
9	3	1	9
1	1	0	1
3	1	3	1

÷ Σ

16	6	7	12

.2	.2	.4	.1	Attention Weight Matrix (A)
.6	.4	.2	.7	
0	.2	0	.1	
.2	.2	.4	.1	

MatMul

	z_1	z_2	z_3	z_4	
	8	8	16	4	Attention Weighted Features
	2	4	4	2	
	6	4	2	7	

FFN

12. Dropout

Random Sequence

.61 >.5
.39
.75
.40
.65 >.33
.42
.23
.19
.93
.42
.87
.53
.27
.69
.50
.11
.42

Training Data:

X_1	X_2
3	5
4	1
1	1

Linear

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

[≈ ReLU]

3	5
7	6
5	2
~~-1~~	4
1	1

Dropout (p=0.5)

$\frac{1}{1-p}=2$

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

[≈ ReLU]

6	10
10	4
~~-1~~	6
1	1

Dropout (p=0.33)

$\frac{1}{1-p}=1.5$

1.5	0	0
0	1.5	0
0	0	~~0~~

9	15
15	6
0	0
1	1

Linear

1	-1	0	0
0	1	-1	-2

-6	9
13	4

Outputs Y

Inference

Unseen Data:

3	3
2	1
1	1

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

[≈ ReLU]

3	3
6	5
0	~~-1~~
1	2
1	1

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

3	3
6	5
0	0
1	2

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

[≈ ReLU]

4	5
9	8
4	4
1	1

1	0	0
0	1	0
0	0	1

4	5
9	8
4	4

1	1	0	0
0	1	-1	-1

13	13
4	3

Training

-4	7
10	5

Targets Y'

(-)

-2	2
3	-1

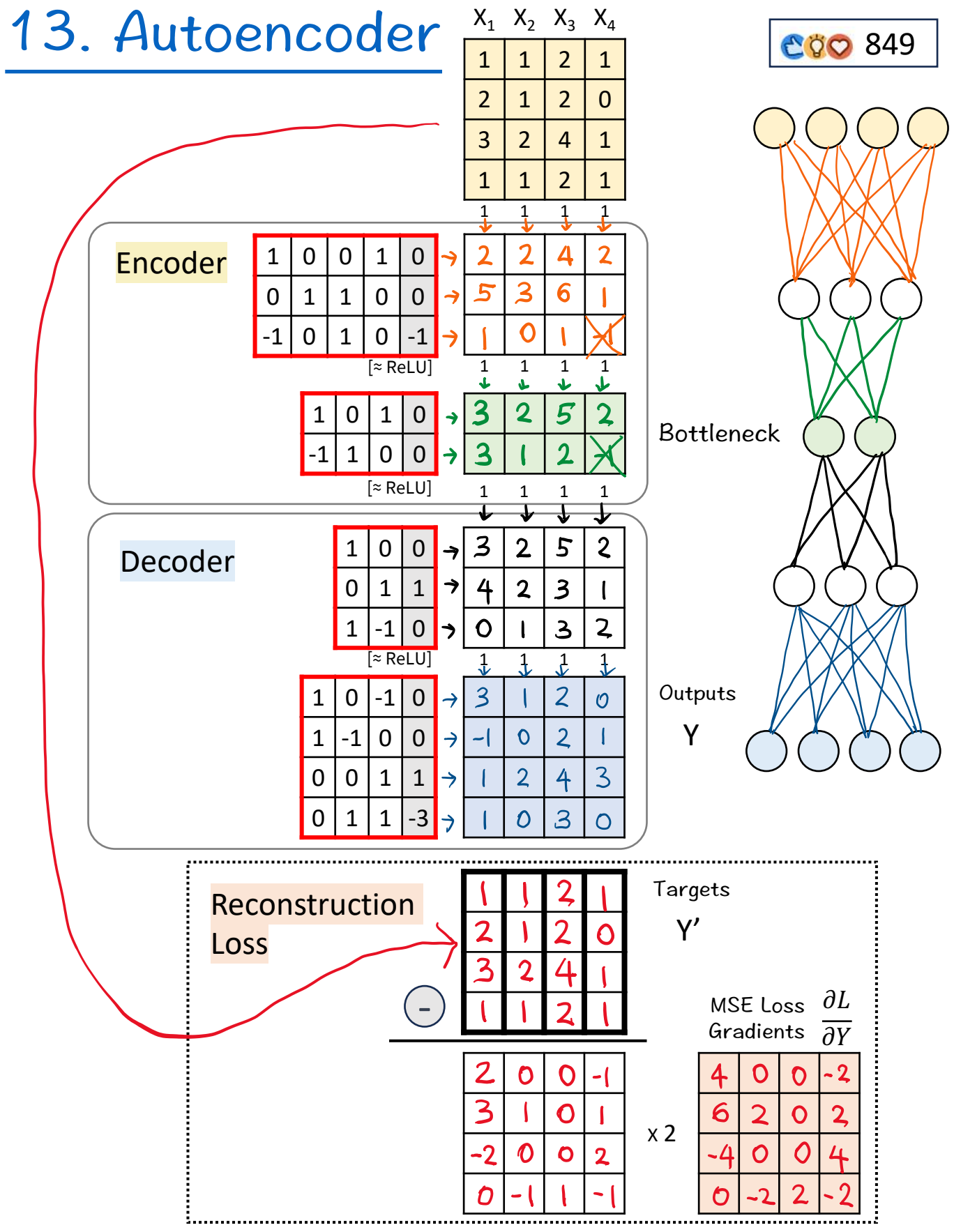
x 2

-4	4
6	-2

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

13. Autoencoder

849



14. Vector Database

2224

Data: how are you | who are you | who am I

Query: 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

Mean Pooling

$\Sigma/3$

Indexing

Projection

Vector Storage

Retrieval

Dot Products: 44/9, 40/9, 60/9

Nearest Neighbor (argmax)

who am I

15. CLIP

885

mini batch of text-image pairs

400 millions more ⋯

big	table			mini	chair			top	hat			table	top			big	chair		

word2vec

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

Flatten Patches

Word Embeddings

Image Encoder

Text Encoder

[Mean Pooling] (round)

[Projection]

Shared Embedding Space

[Softmax]

Similarity Image→Text

Target

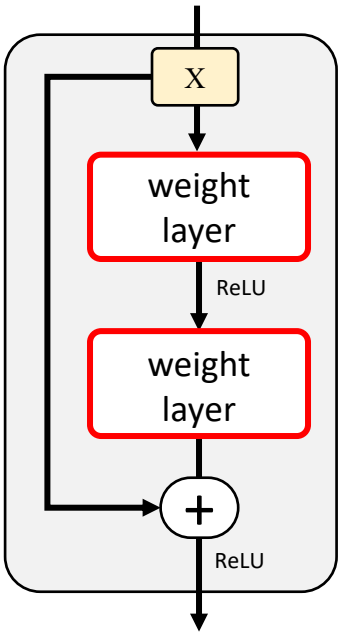
Cross Entropy Loss Gradients

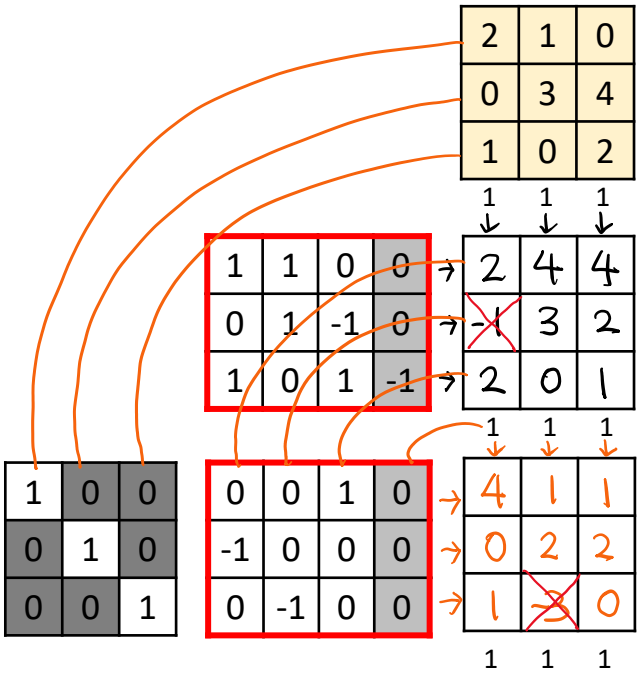
Image→Text

Text→Image

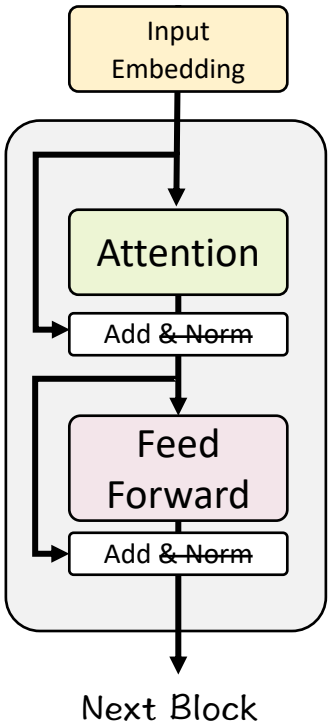
Similarity Text→Image

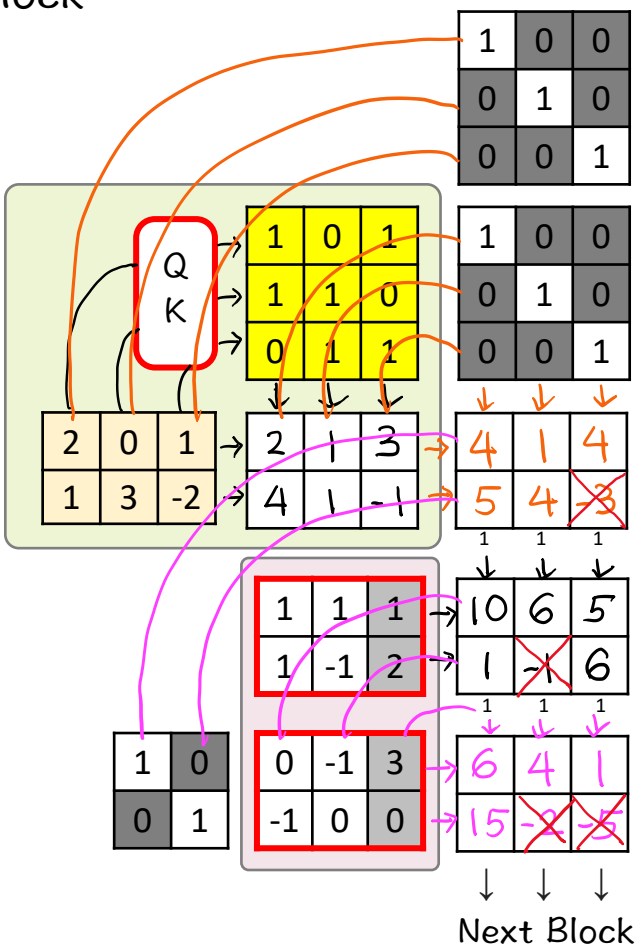
16. Residual Network





Transformer's Encoder Block

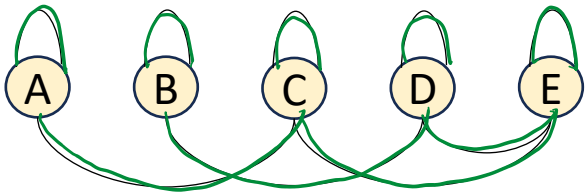




17. Graph Convolutional Network

573

Graph Data



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

Adjacency Matrix

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

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

[ReLU]

Messages

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

[ReLU]

Messages

Fully Connected Network

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

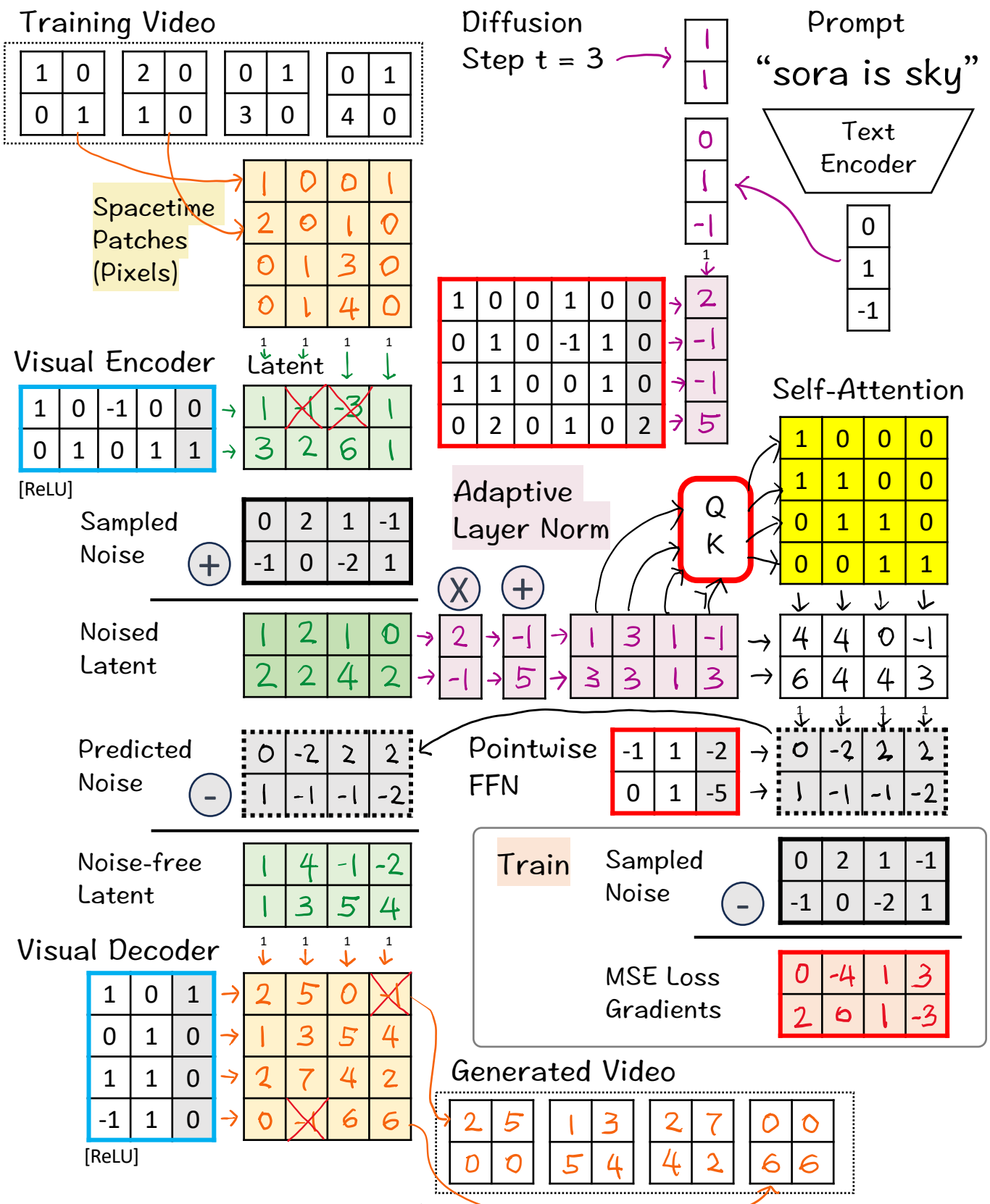
[ReLU]

1	1	1	1	1	-9

σ

18. SORA's Diffusion Transformer

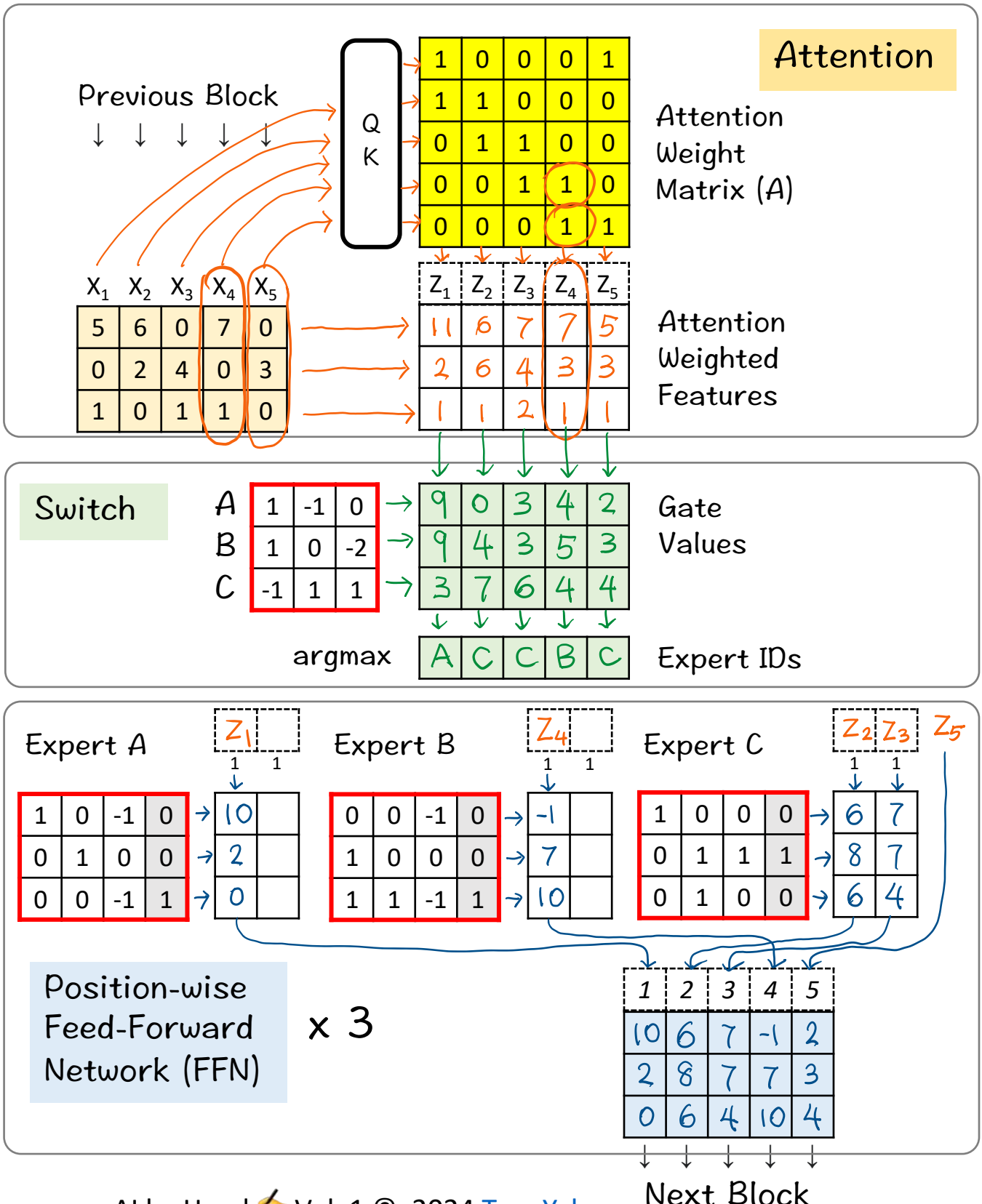
2118



19. Switch Transformer

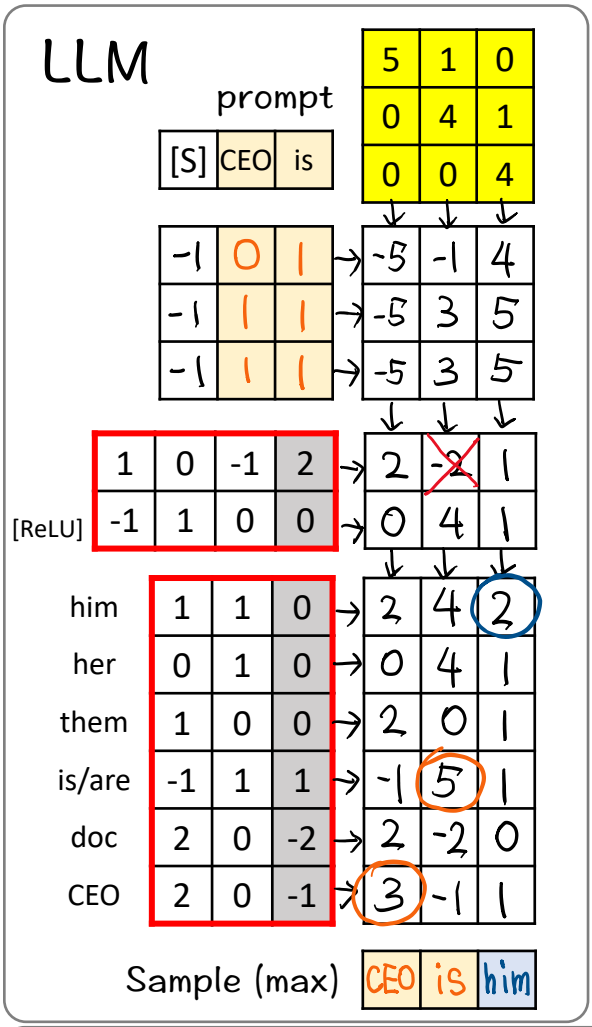
576

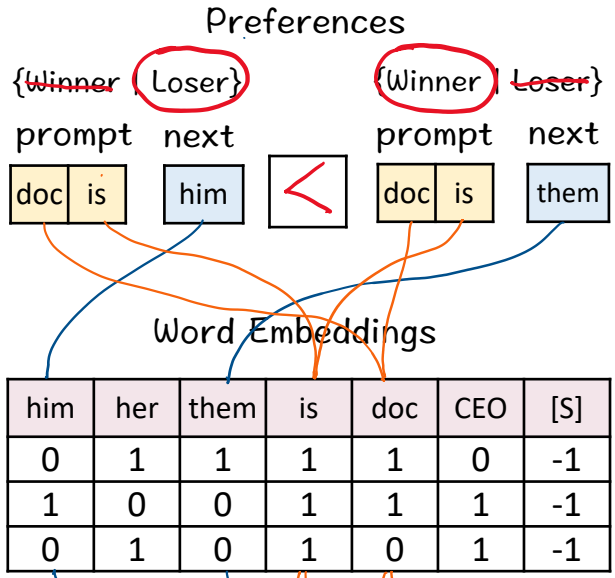
(Gemini 1.5's Sparse Mixture of Experts)

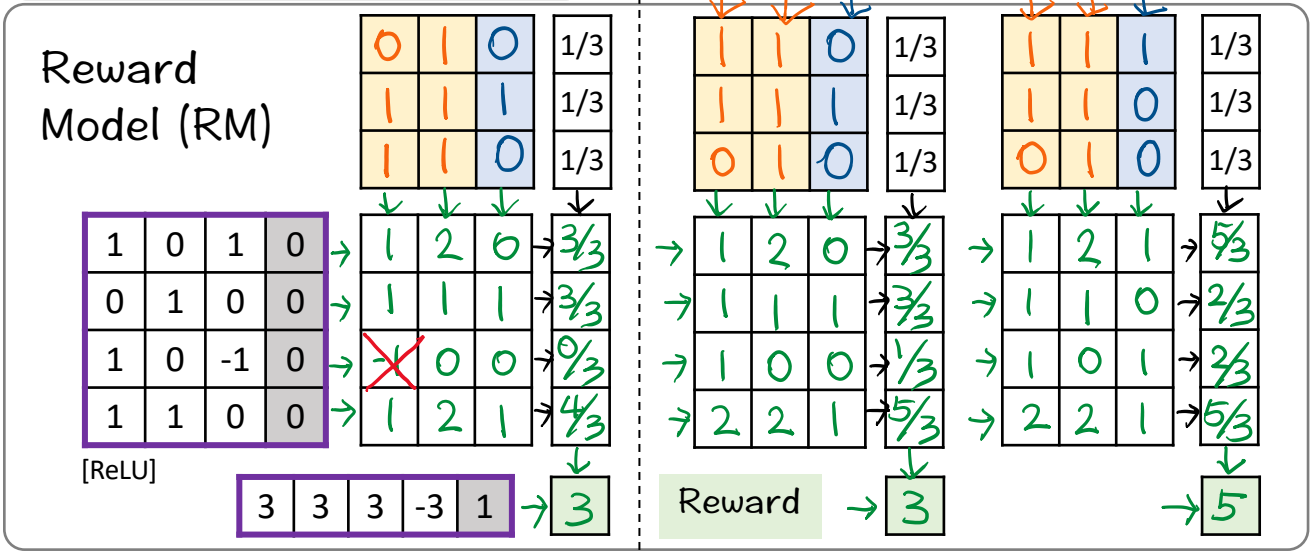


20. Reinforcement Learning with Human Feedback (RLHF)

550







Align LLM

Loss -3

Loss Gradient -1

Train RM

Winner - Loser 5 - 3 = 2

Predicted σ 0.9

- Target -1

Loss Gradient -0.1