

AI by Hand



Volume 1

Basic x 5
Advanced x 20

Prof. Tom Yeh

2024

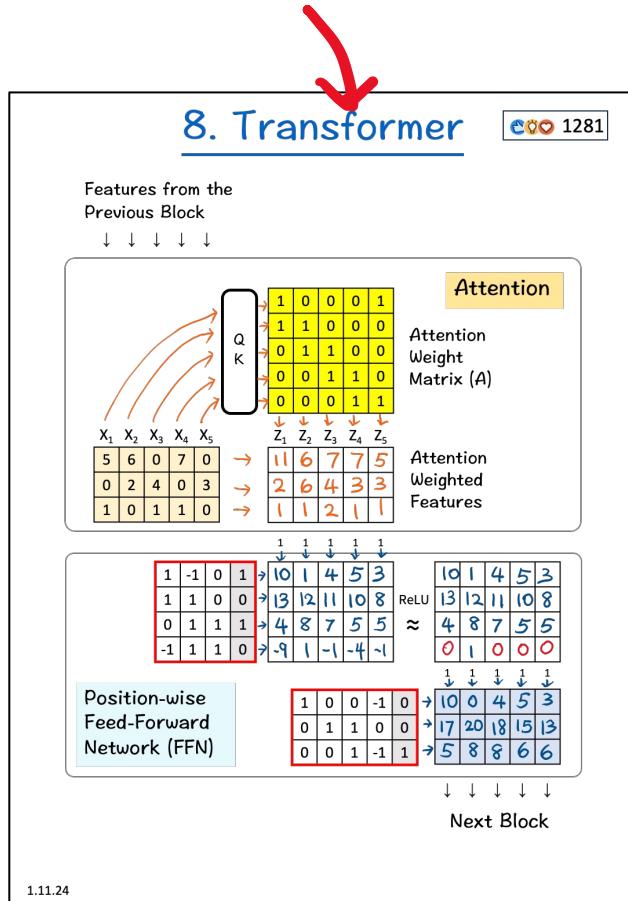
Basic

- 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

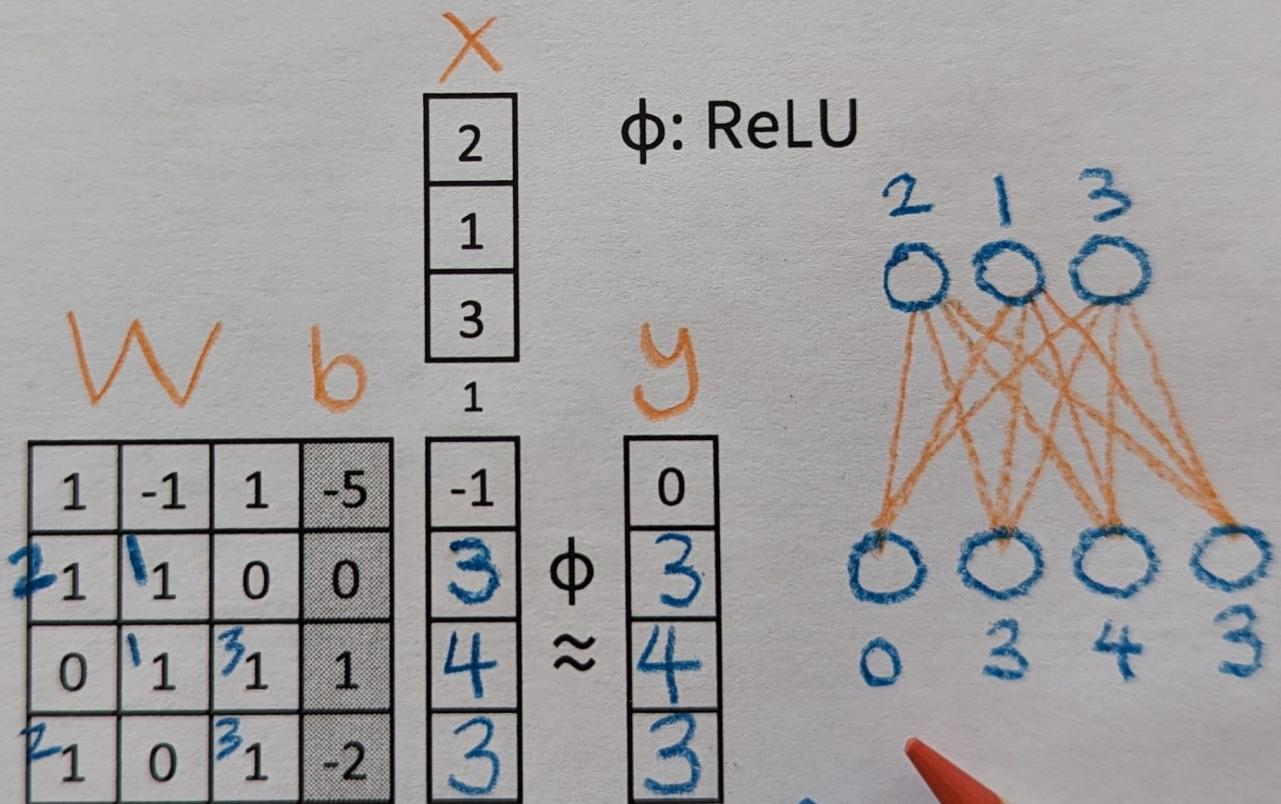
$$\begin{array}{c} \boxed{2} \\ \boxed{1} \\ \boxed{3} \end{array} \times \quad \text{ReLU}$$

w b 1 ϕ \approx 0

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

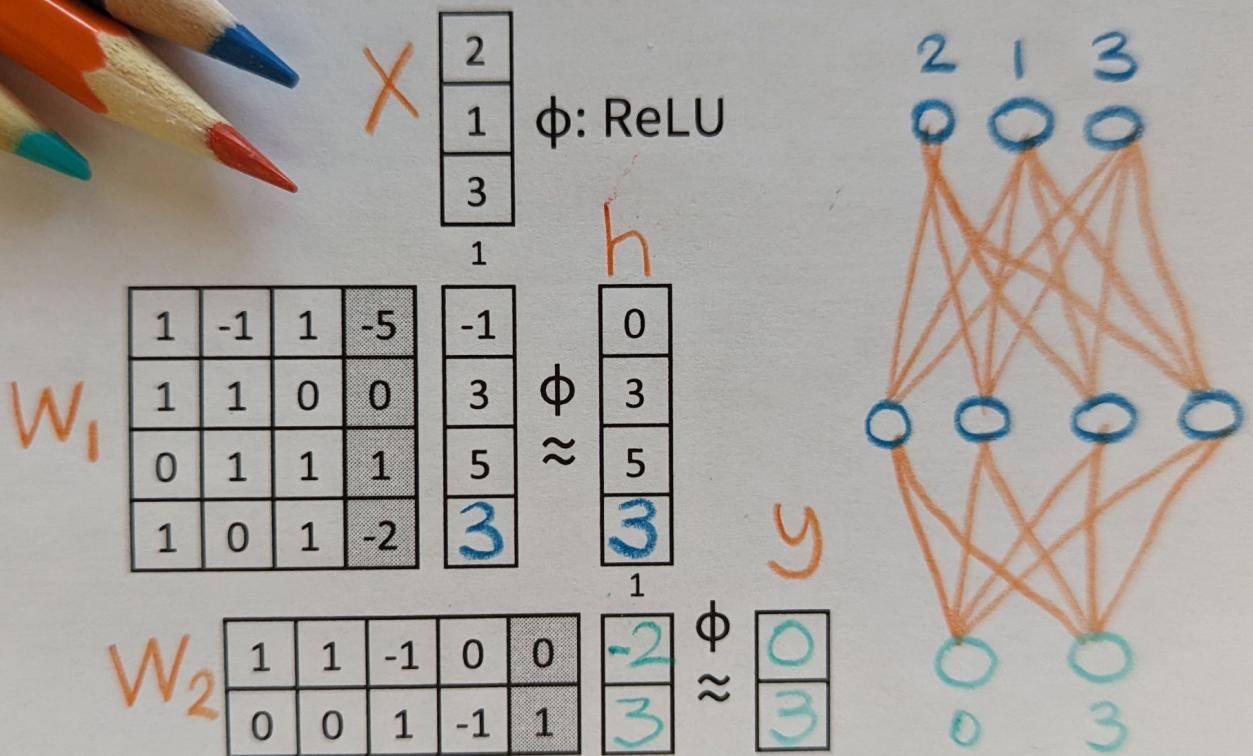
© 2023 Tom Yeh.

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

ϕ : 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

$\phi \approx$

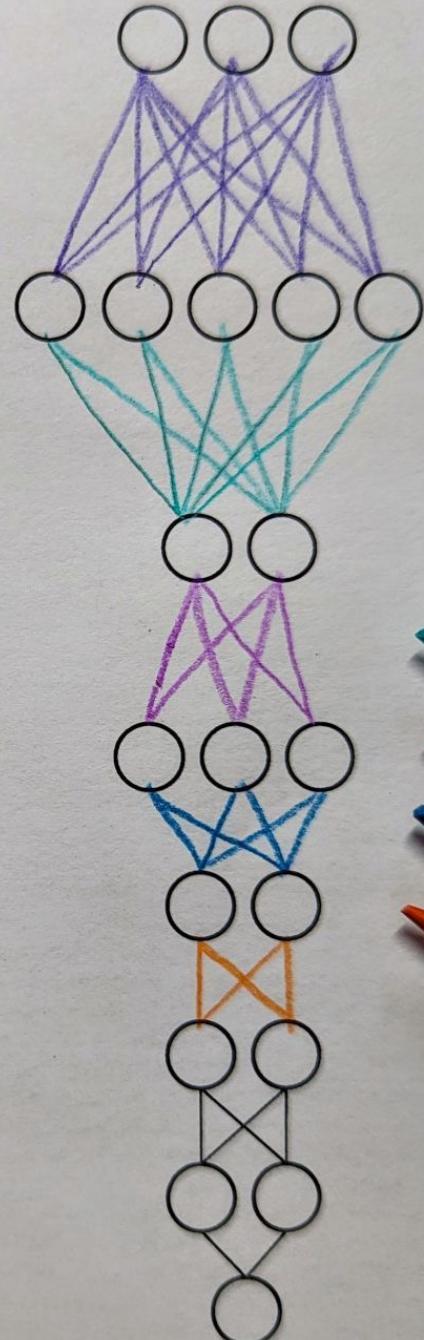
0	0	0
3	3	4



© 2023 Tom Yeh.

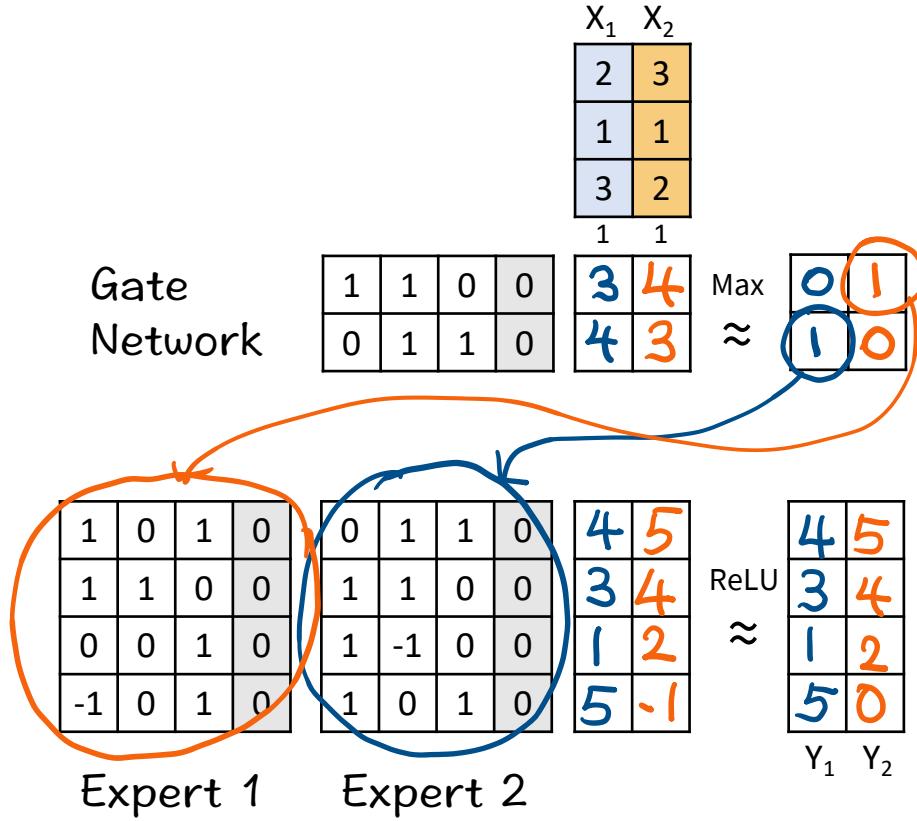
V. Seven Layers

					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 -5
	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

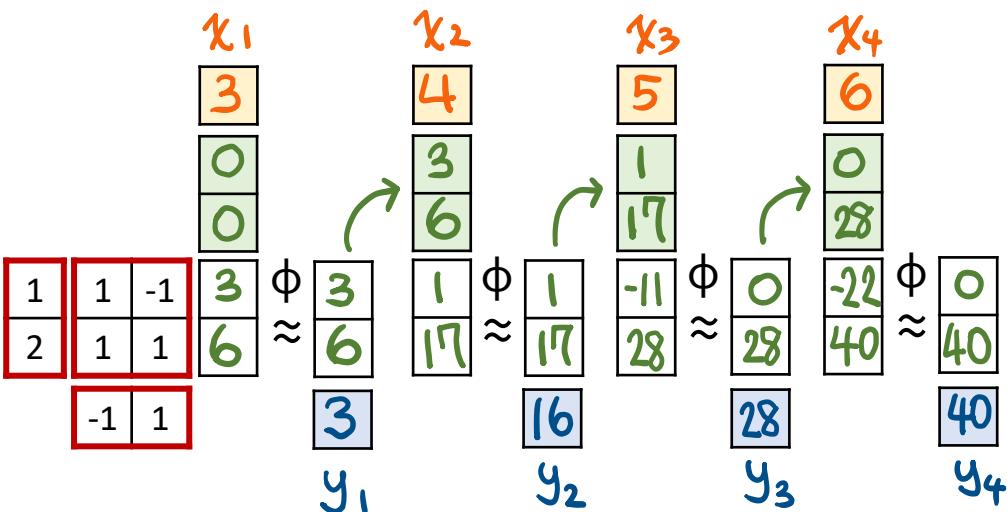
ϕ : ReLU

Hidden States

H_0	0	
	0	

Output Sequence

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

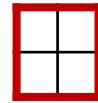


3. Mamba's S6 Model

Input Sequence

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

Parameters

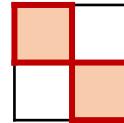


Output Sequence

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

Selective

Structured



State-Space

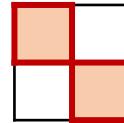


1	-1	0	0
0	-1	0	1
1	0	-1	0
1	0	0	-1
1	0	-1	0
0	1	0	-1
1	-1	0	0
0	0	-1	1
-1	0	0	0
1	0	0	0
0	0	-1	0
0	1	0	0
1	-1	0	0
0	0	-1	1
1	0	0	0
0	-1	1	0

3
4
5
6

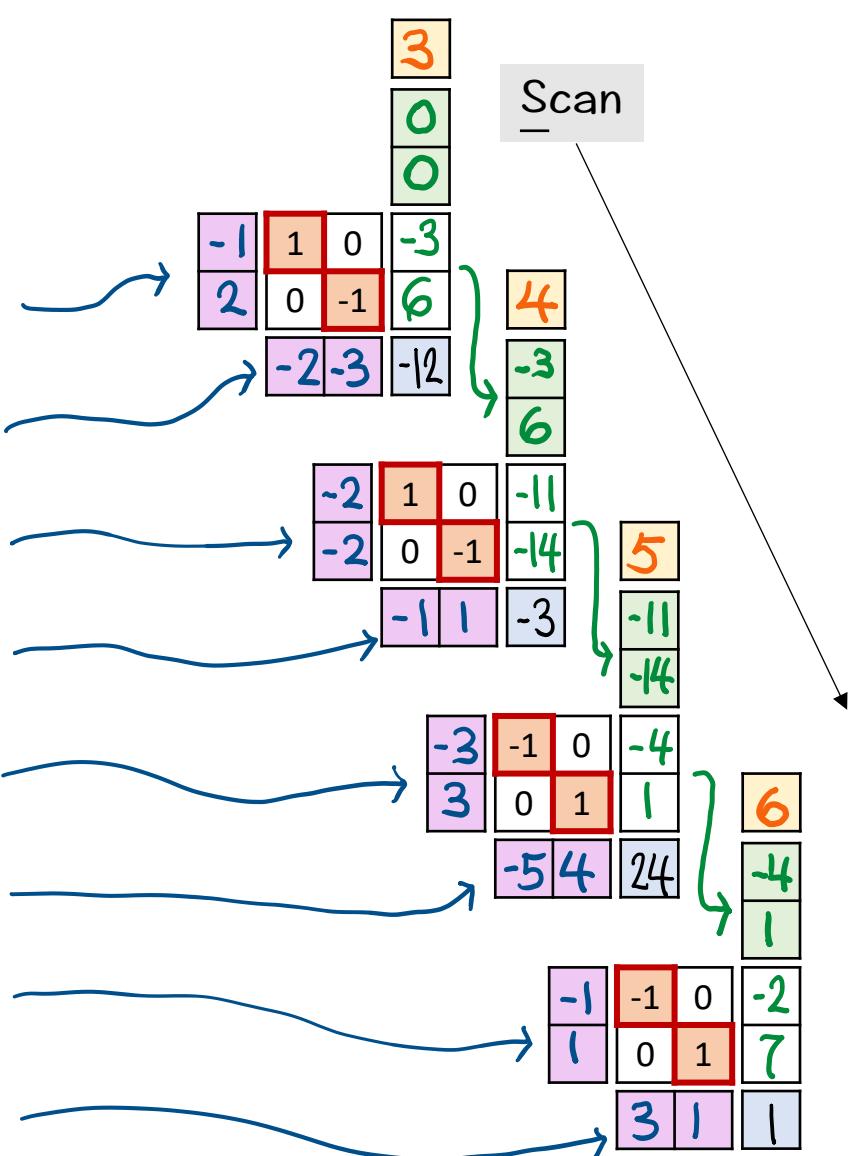
-1
2
-2
-3
-2
-2
-1
1
-3
3
-5
4
-1
1
3
1

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



3
0
0
6
-3
-12
-11
-14
-3
-11
-14
5
-4
1
24
-4
1
6
4
1

Scan



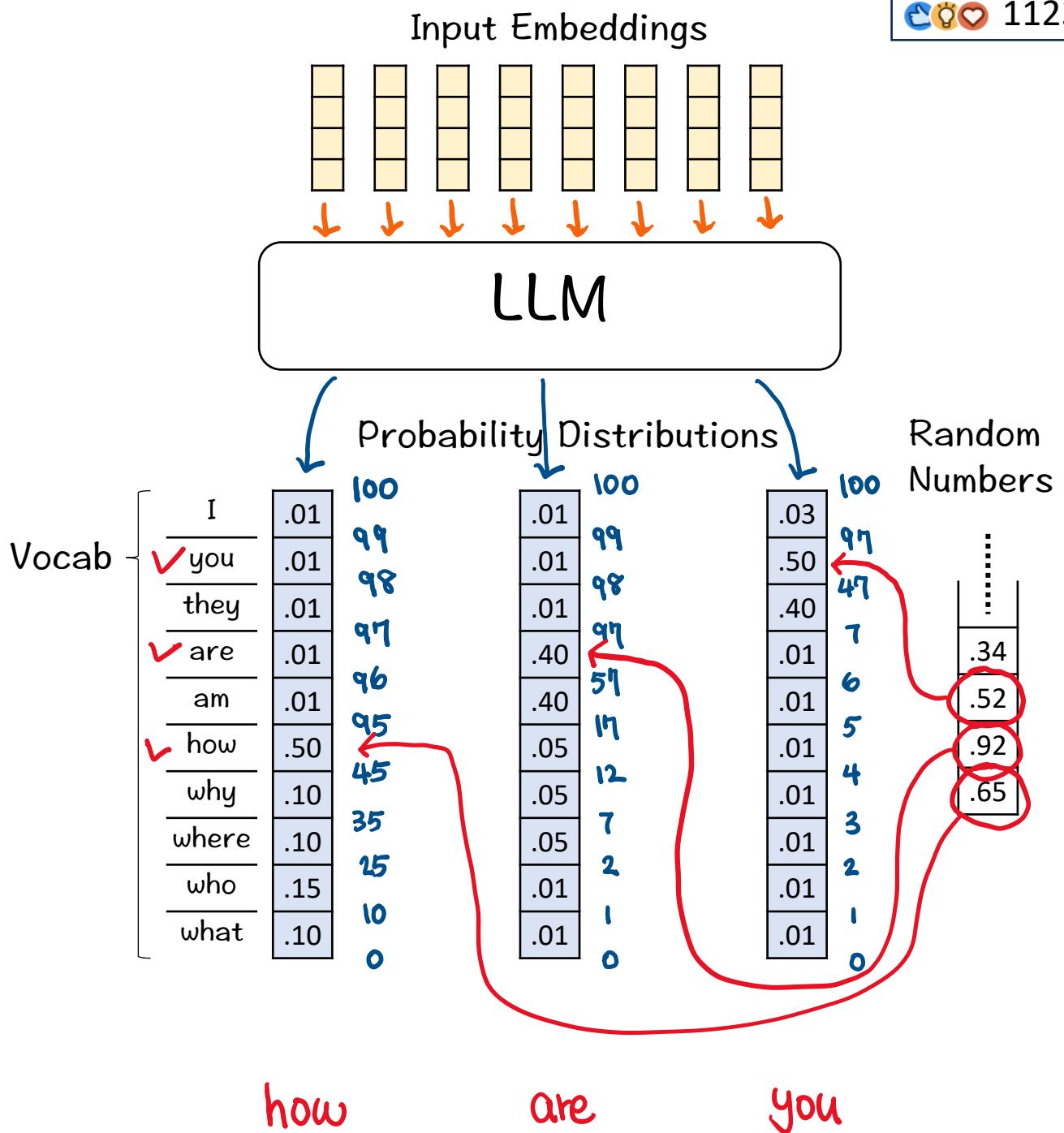
4. Matrix Multiplication

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

1	5	2
2	4	2

1	1	-1
3	9	4
1	-1	0

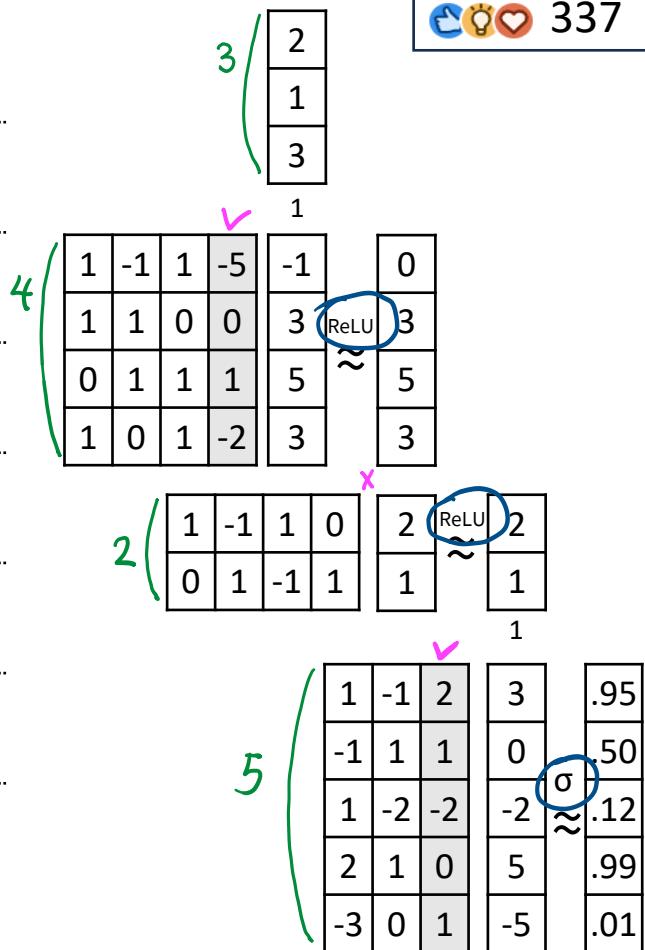
5. How does an LLM sample a sentence?



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 }

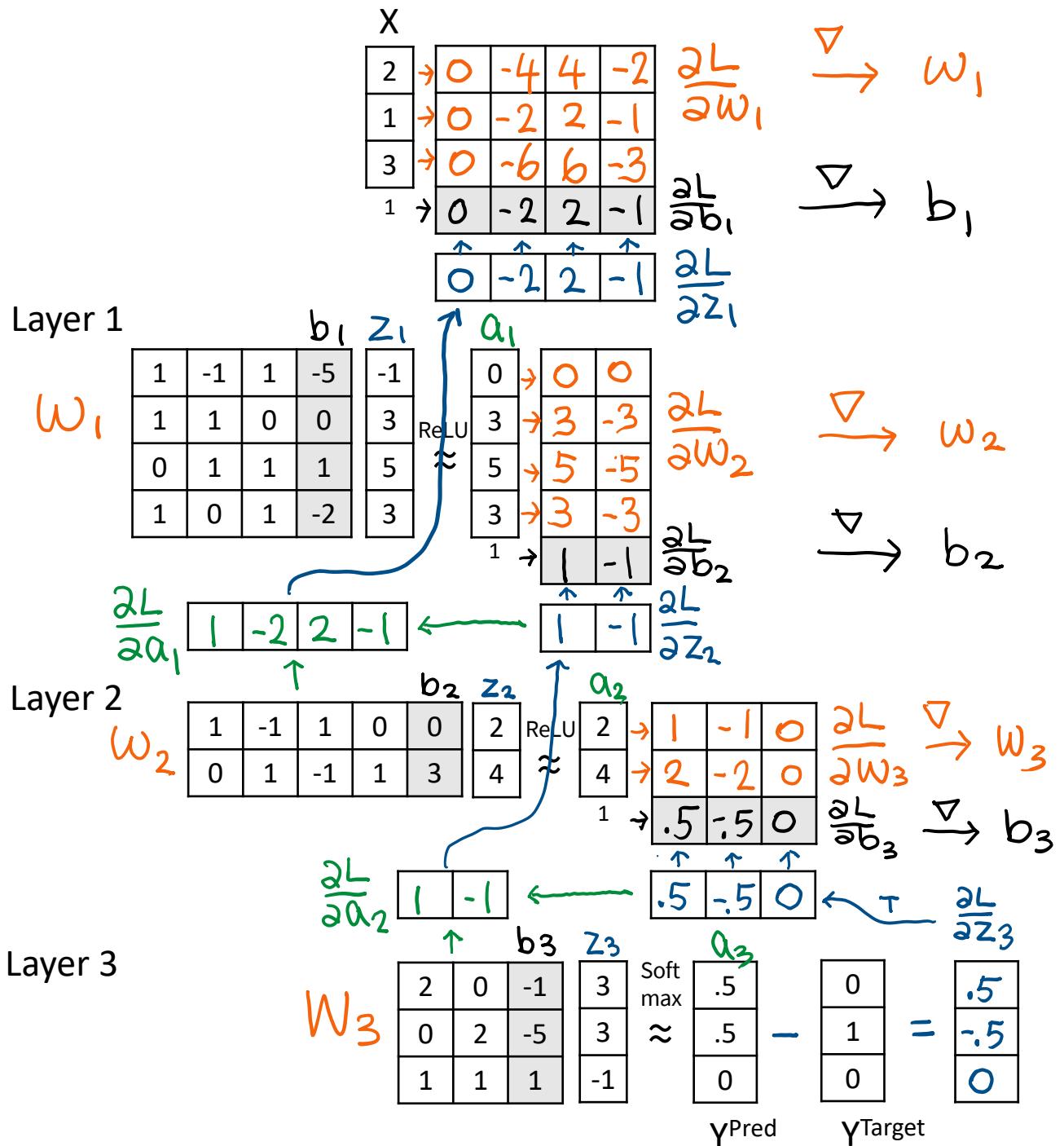
in_features: { int }

out_features: { int }

bias: { T | F }

7. Backpropagation

1260

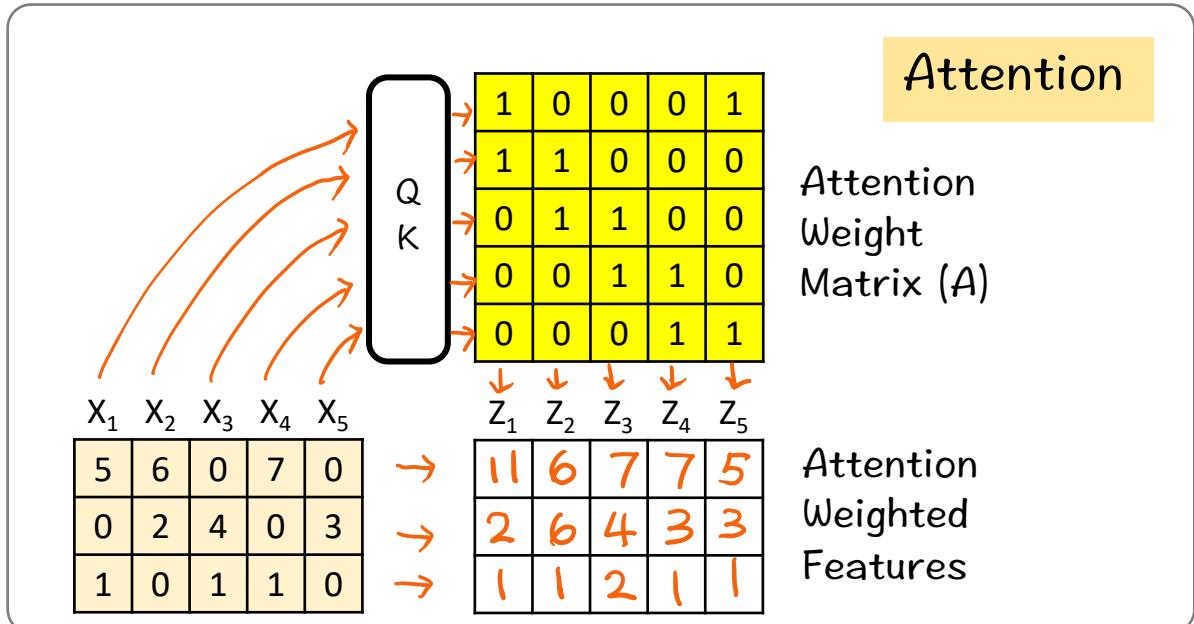


L : Cross-Entropy Loss

8. Transformer

Features from the
Previous Block

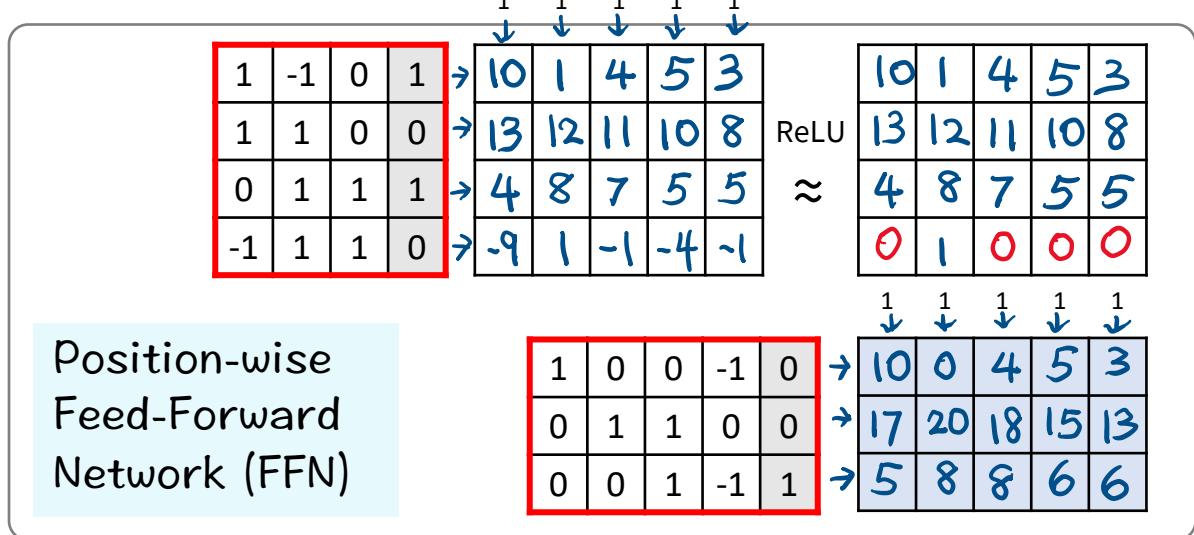
↓ ↓ ↓ ↓ ↓



Attention

Attention
Weight
Matrix (A)

Attention
Weighted
Features



Position-wise
Feed-Forward
Network (FFN)

ReLU
≈

10	1	4	5	3
13	12	11	10	8
4	8	7	5	5
0	1	0	0	0

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

10	0	4	5	3
17	20	18	15	13
5	8	8	6	6

↓ ↓ ↓ ↓ ↓

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

Σ	μ	σ^2	σ
9	2	1	1
5	1	1	1
7	2	4	2

Normalize

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

Sum (Σ)
Mean (μ)
Variance (σ^2)
Std Dev (σ)

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

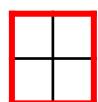
$$\begin{array}{r} \sigma \\ \div \\ \hline 1 \\ 1 \\ 2 \end{array}$$

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 ₁ N ₂ N ₃ N ₄			
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 ₁	1 1 2 1
F ₂	2 1 2 0
F ₃	3 2 4 1
F ₄	1 1 2 1

[≈ ReLU]

Real:

X ₁	X ₂	X ₃	X ₄
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 _D	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 _G	1 1 1 1
---	----------------	---------------

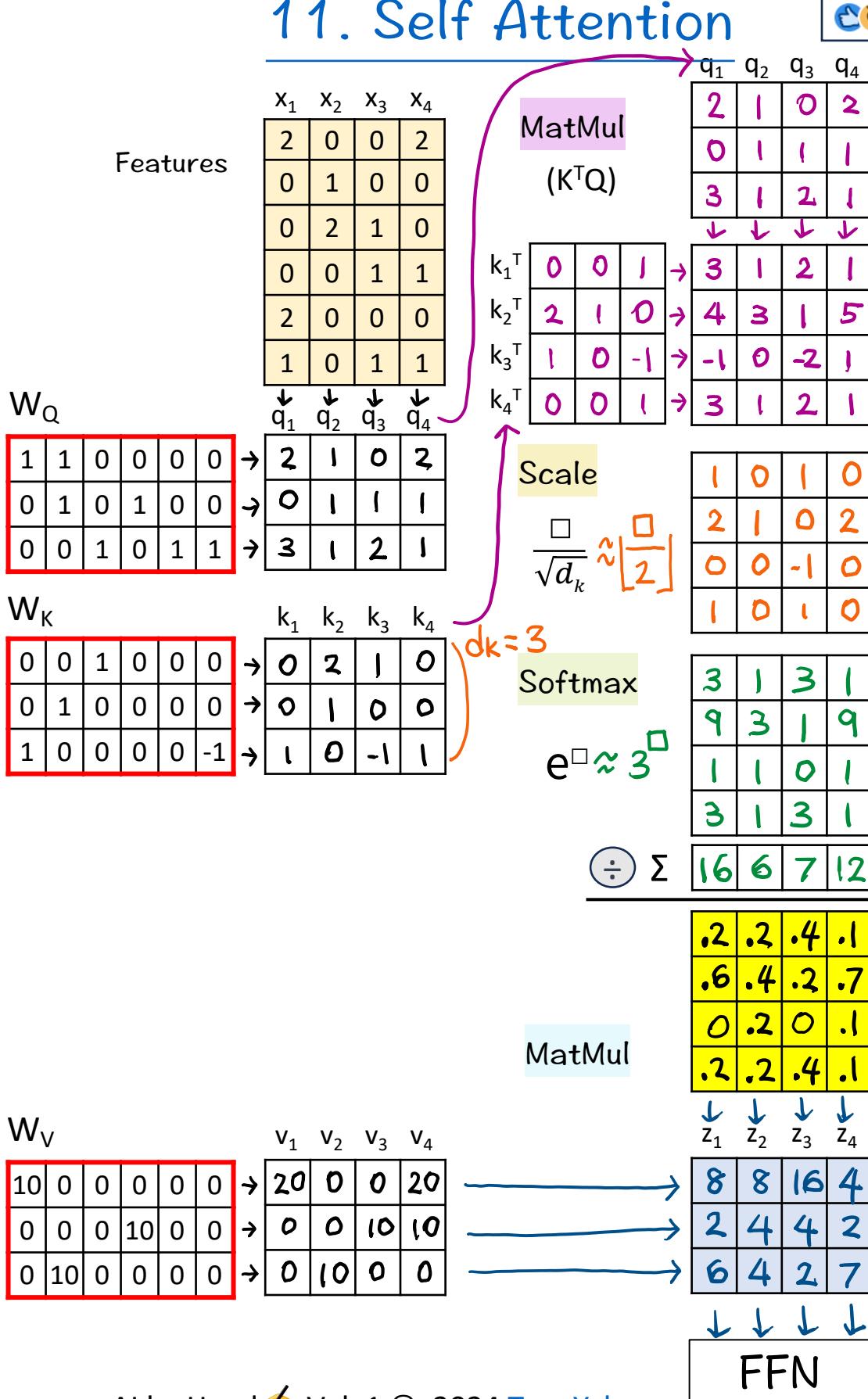
Loss Gradients: $\frac{\partial L_G}{\partial Z}$

-3	-5	-1	-7
----	----	----	----

11. Self Attention



1376



12. Dropout



557

Random Sequence

.61 >.5

.39
.75
40
.65
.42
.23

Linear

Training Data:

X_1	X_2
3	5
4	1

1 1

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

[≈ ReLU]

3	5
7	6

5 2

4

Dropout
($p=0.5$)

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

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

4

6

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

Training



-4	7
10	5

-2	2
3	-1

Targets

Y'

-4	4
6	-2

$\times 2$

Inference

Unseen Data:

3	3
2	1

1 1

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

[≈ ReLU]

3	3
6	5

0 X

1 2

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

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	0	0	0
0	1	0	0
0	0	0	1

4	5
9	8

4 4

1	1	0	0
0	1	-1	-1

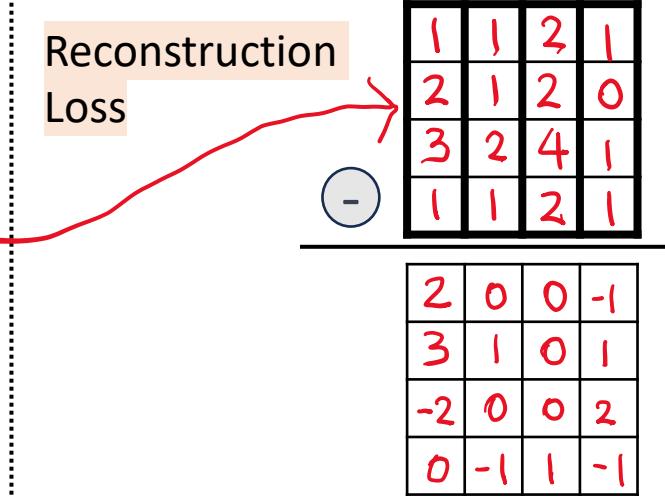
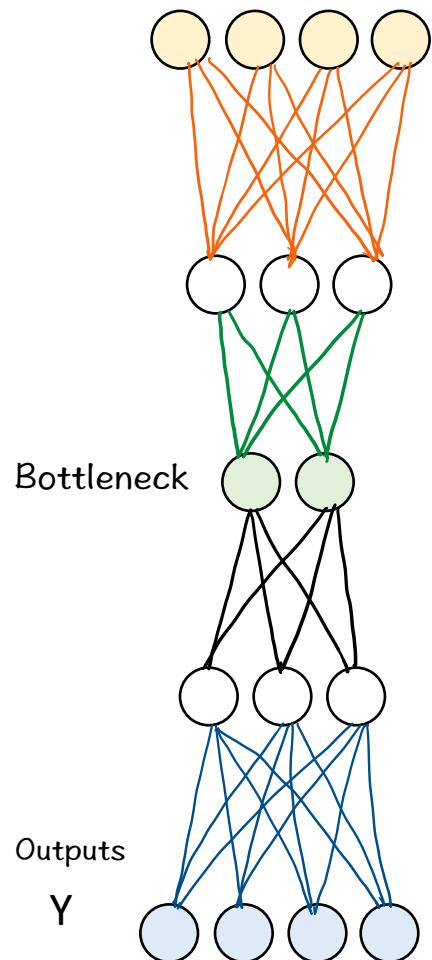
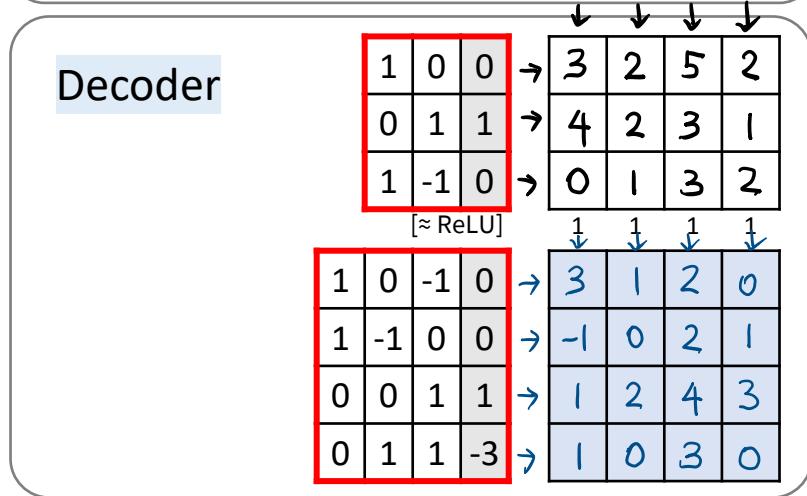
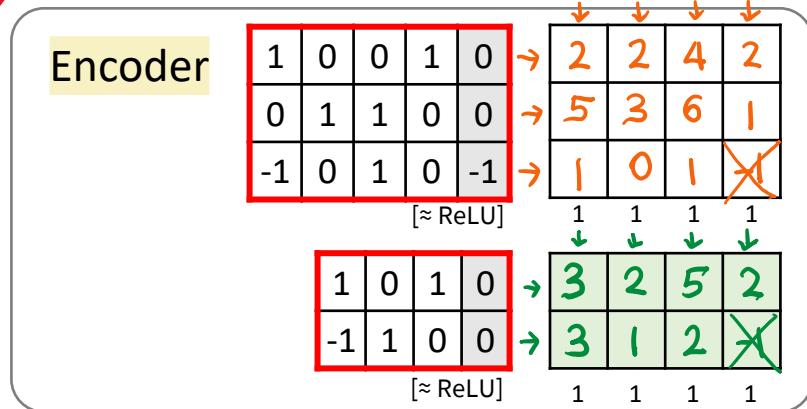
13	13
4	3

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

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

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

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

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

Vector Storage

8/3
2/3

Retrieval

Dot Products

44/9

40/9

60/9

T

Nearest Neighbor (argmax)

✓

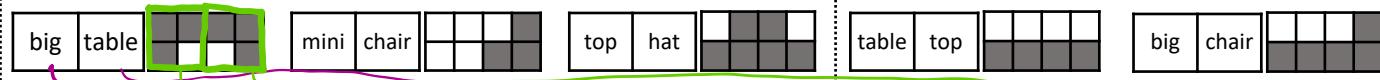
who am I

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

T₁ T₂ T₃

1	0	-1
1	2	0

I ₁	2	0
I ₂	1	0
I ₃	2	1

[Softmax]

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

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

Similarity
Image → Text

$$\frac{.9}{.75}$$

$$\frac{.1}{.25}$$

$$\frac{0}{0}$$

$$\frac{1}{1}$$

Target

$$\frac{1}{0}$$

$$\frac{0}{1}$$

$$\frac{0}{0}$$

Cross Entropy
Loss Gradients

Image → Text

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

Text → Image

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

Similarity
Text → Image

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

Flatten
Patches

1	1
1	1
1	0
0	1
0	1

0	0
0	1
1	0
0	1
0	1

0	1
1	1
1	3
1	1
2	1

Image Encoder

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

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

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

1	1	1
3	2	2
1	1	1
2	1	2
1	2	2

[Mean Pooling]
(round)

2	1	1
1	2	0
2	1	1

[Projection]

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

2	1	1
0	0	1
1	0	1

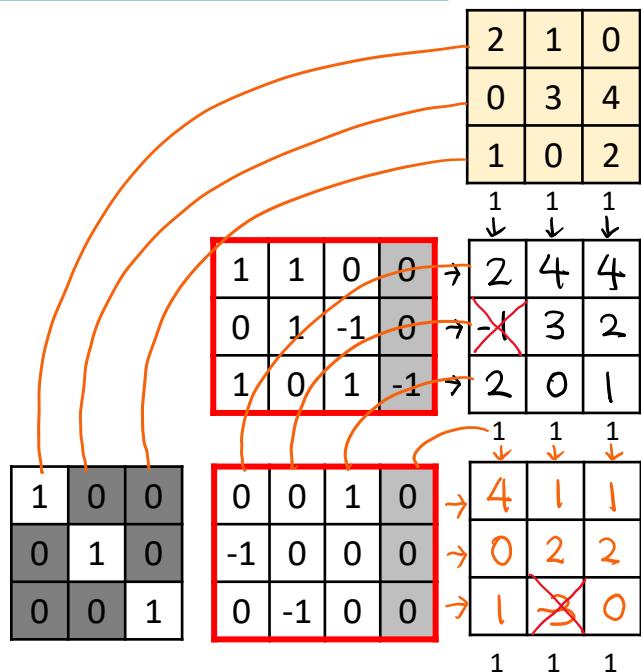
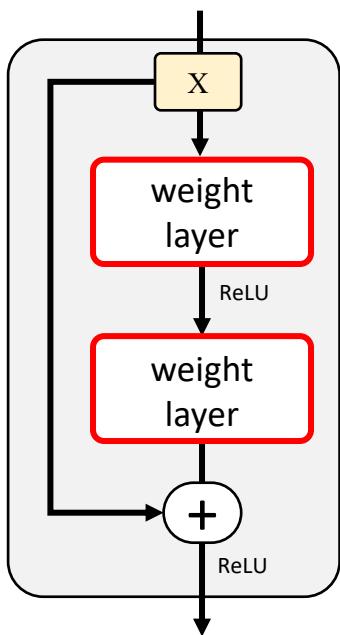
Image → Text

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

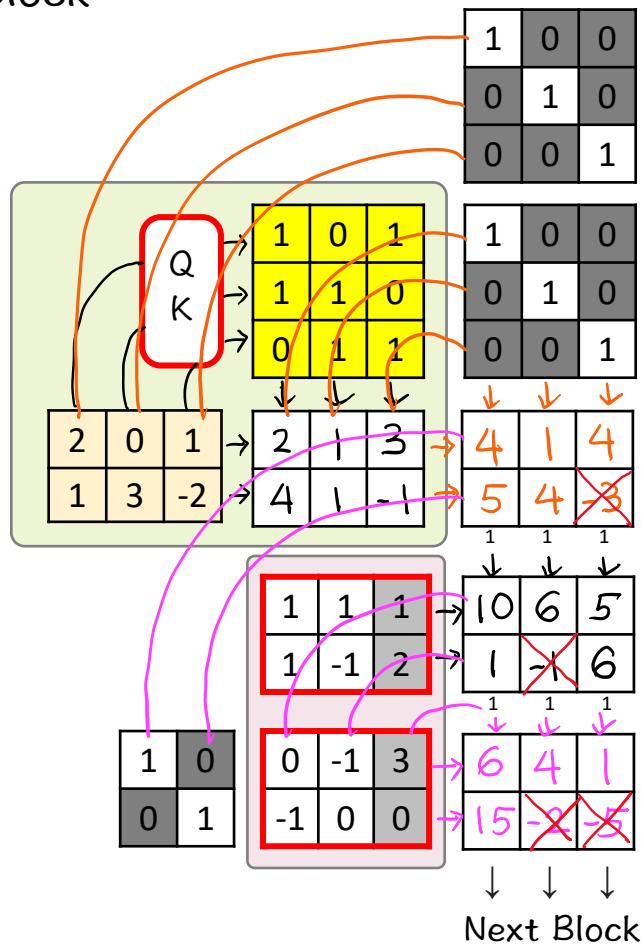
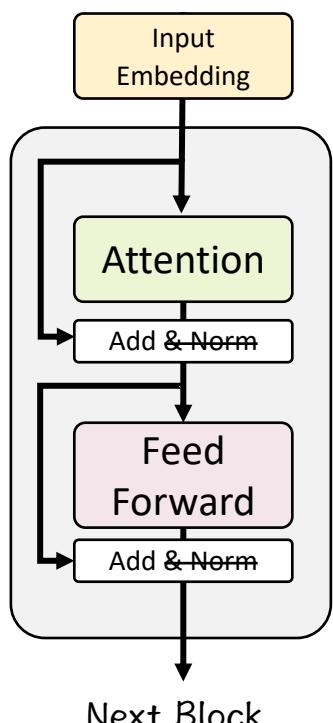
Text → Image

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

16. Residual Network

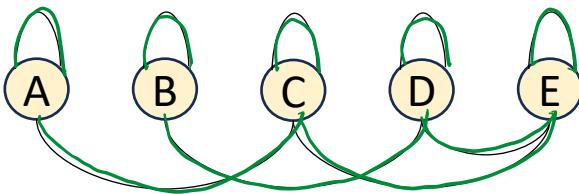


Transformer's Encoder Block



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

→ 4 1 5 2 1

→ 0 1 0 1 0

→ 1 2 1 2 0

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

→ 2 6 2 6 4

→ 5 0 5 0 3

→ 9 3 10 4 8

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
---	---	---	---	---	----

→ -4 4 5 3 0

σ 0 1 1 1 0.5

18. SORA's Diffusion Transformer

 2118

Training Video

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

Spacetime
Patches
(Pixels)

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

Visual Encoder

1 0 -1 0 0
0 1 0 1 1

[ReLU]

Sampled
Noise



0 2 1 -1
-1 0 -2 1

Noised
Latent

1 2 1 0
2 2 4 2

Predicted
Noise



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

Noise-free
Latent

1 4 -1 -2
1 3 5 4

Visual Decoder

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

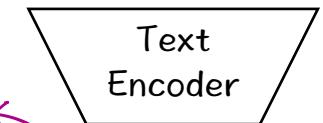
[ReLU]

Diffusion

Step $t = 3$

1
1

Prompt
“sora is sky”



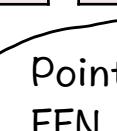
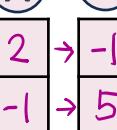
0
1
-1

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

Self-Attention

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

Adaptive
Layer Norm



Pointwise
FFN

-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

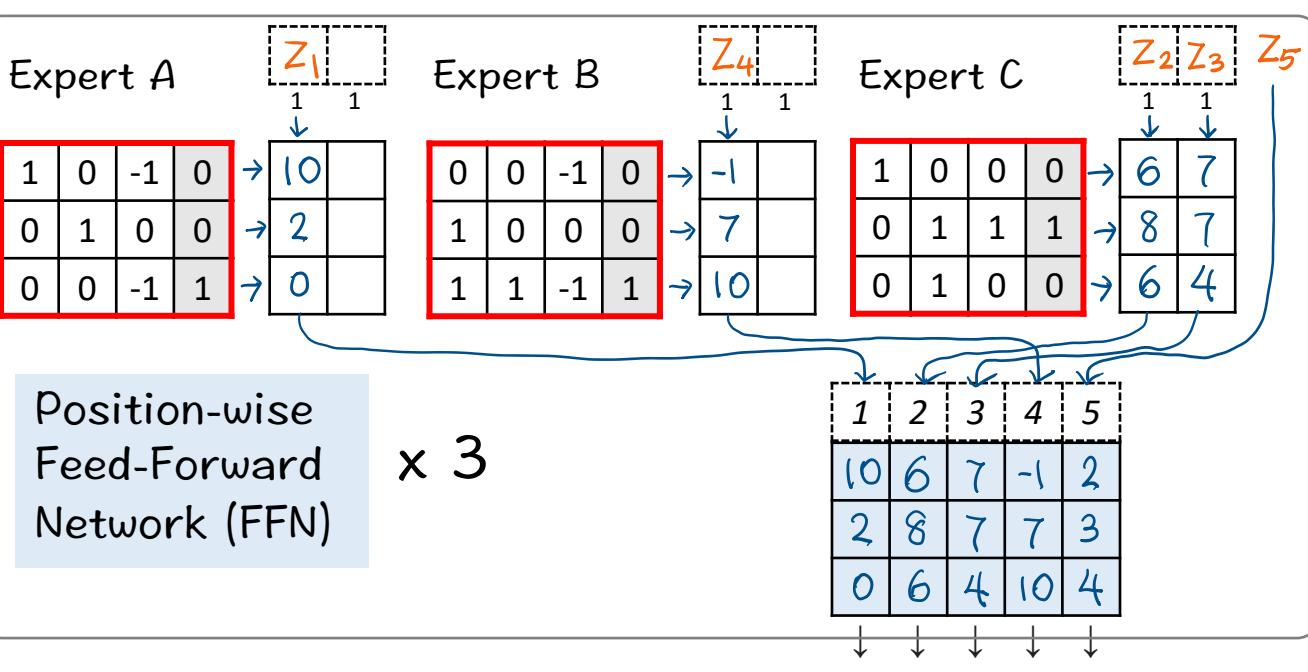
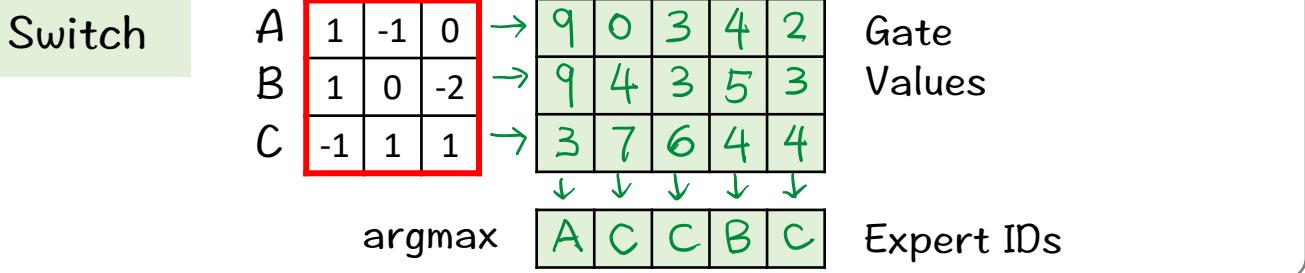
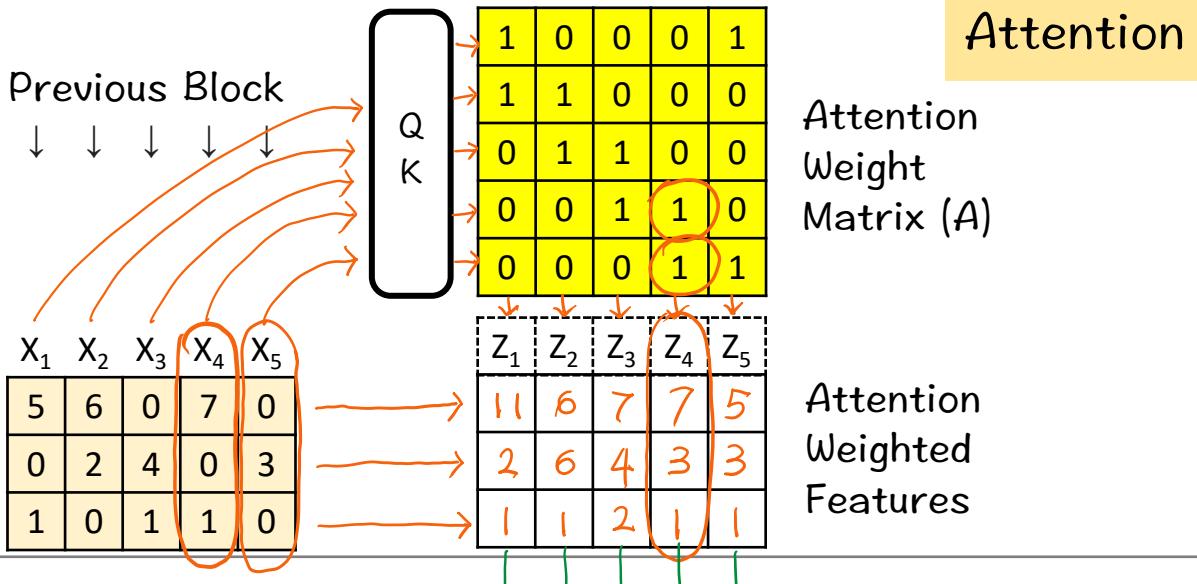
2 5
1 3
2 7
0 0

19. Switch Transformer



576

(Gemini 1.5's Sparse Mixture of Experts)



20. Reinforcement Learning with Human Feedback (RLHF)

