LLAMA x 3

Tom

University of Colorado Boulder

AI by Hand ✍

Ravi

LlamaIndex

RAG

Eduardo

aws

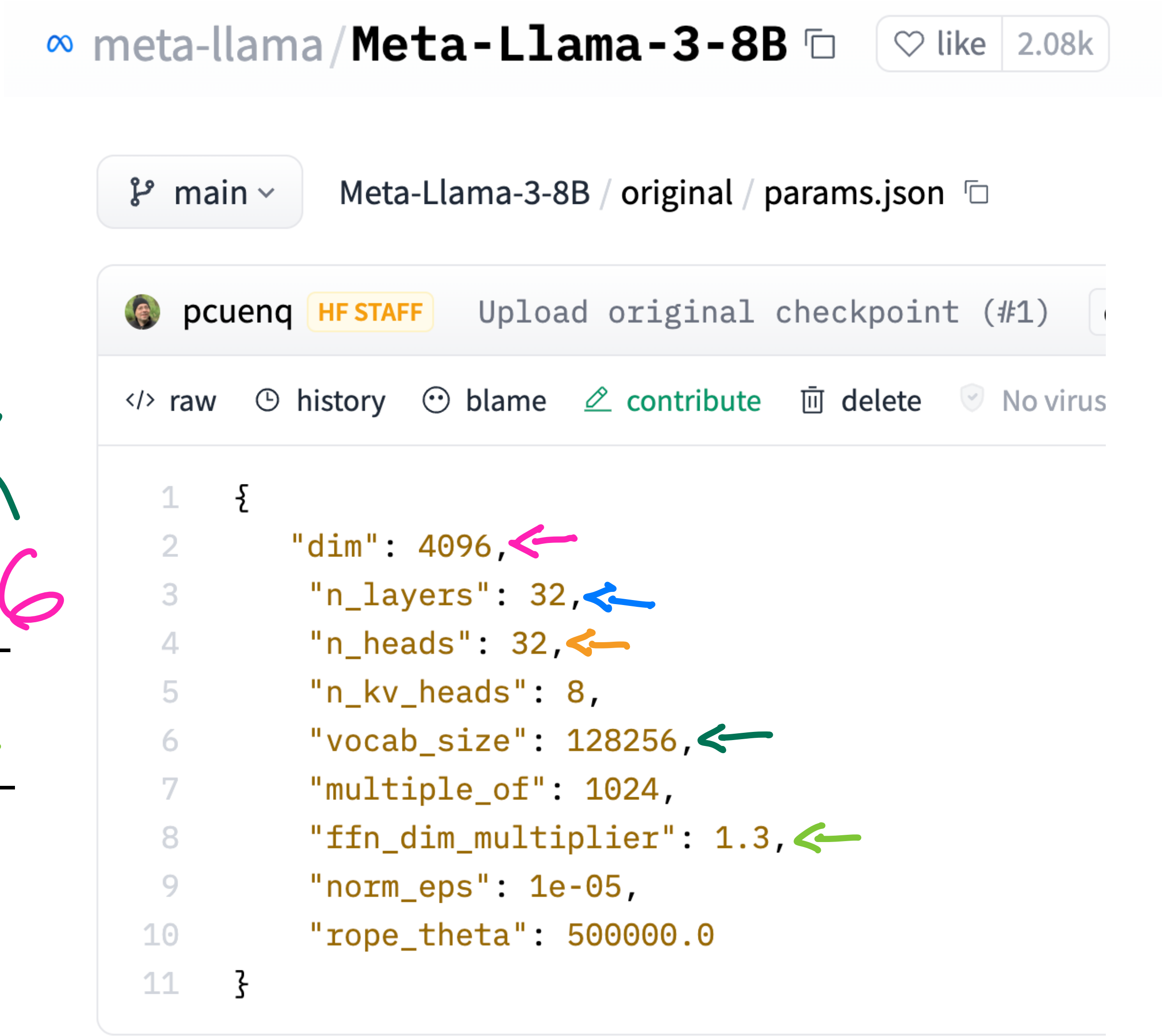
MLOps

LLaMA 3

What are the dimensions?

8B model parameters

- # of Layers = 32
- # of Attention Heads = 32
- # of Vocabulary Words = 128K
- # of Feature Dimensions = 4096
- # of Hidden Dimensions = 5012
- Context Window Size = 8K



meta-llama / Meta-Llama-3-8B

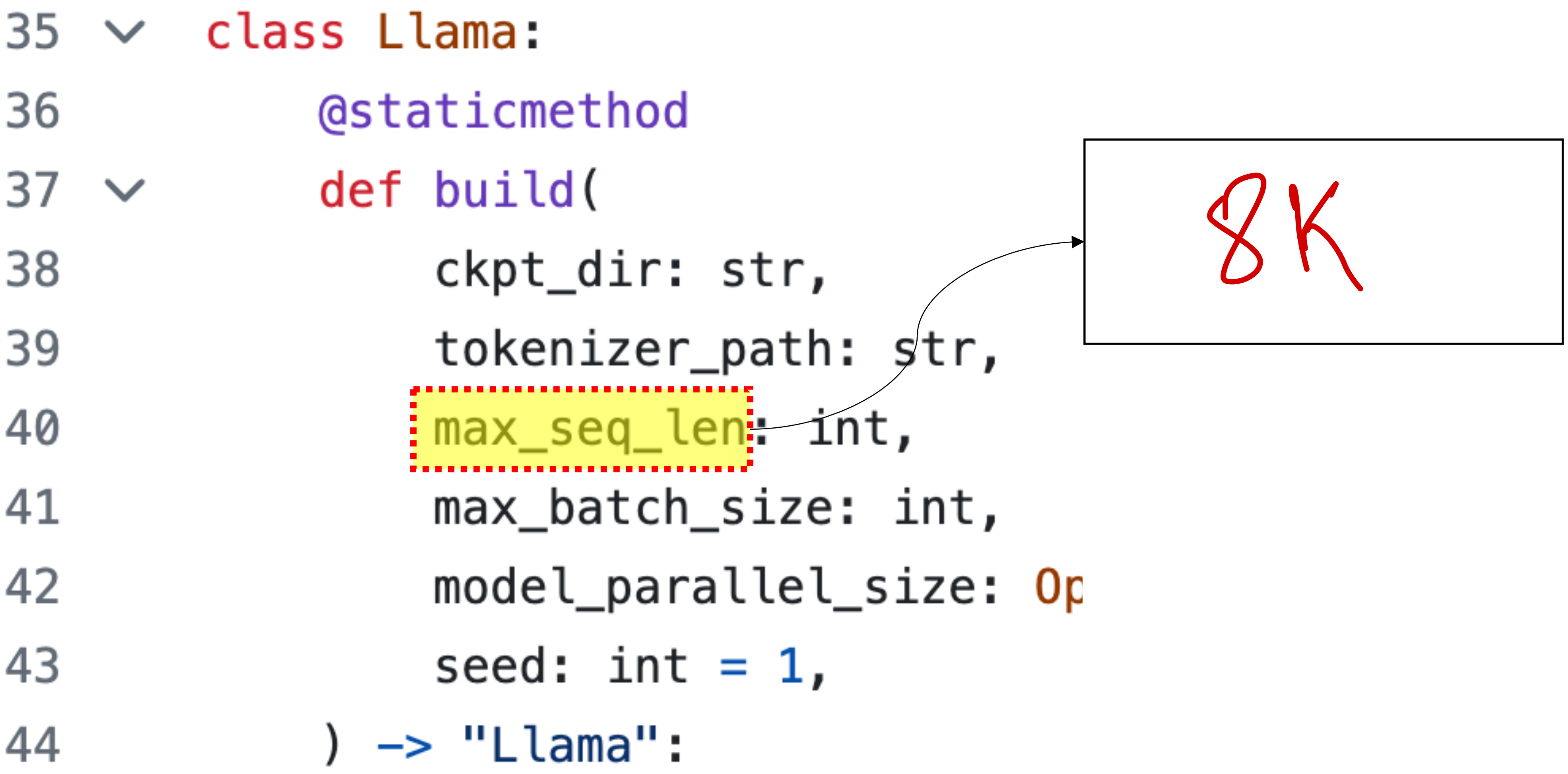
Meta-Llama-3-8B / original / params.json

pcuenq HF STAFF Upload original checkpoint (#1)

```
{
    "dim": 4096,
     "n_layers": 32,
     "n_heads": 32,
     "n_kv_heads": 8,
     "vocab_size": 128256,
     "multiple_of": 1024,
     "ffn_dim_multiplier": 1.3,
     "norm_eps": 1e-05,
     "rope_theta": 500000.0
}
```

$4096 \cdot 1.3 = \underline{5012}$

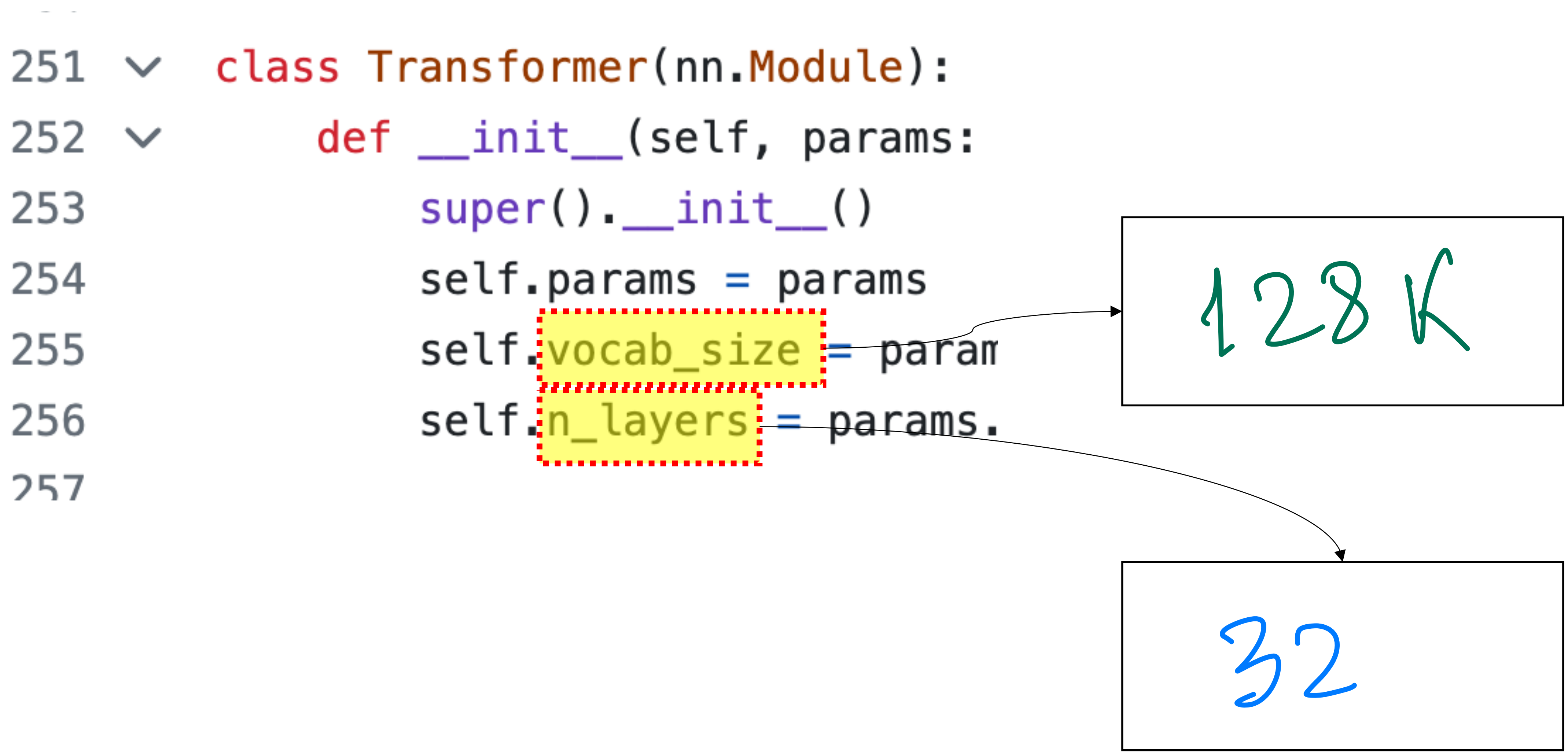
Llama



```python
class Llama:
    @staticmethod
    def build(
        ckpt_dir: str,
        tokenizer_path: str,
        max_seq_len: int,
        max_batch_size: int,
        model_parallel_size: Op
        seed: int = 1,
    ) -> "Llama":
```

8K

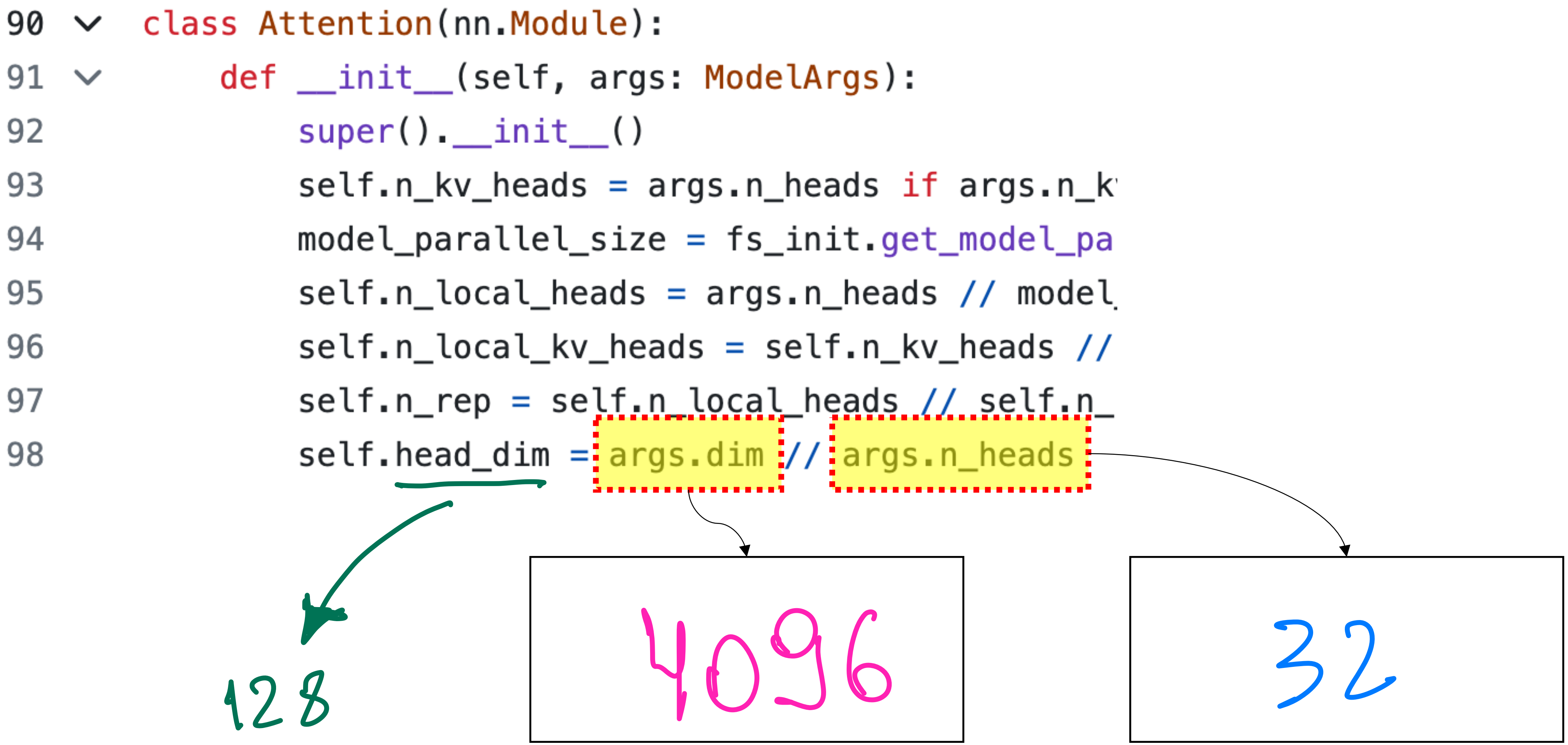
Transformer



```
class Transformer(nn.Module):
    def __init__(self, params:
        super().__init__()
        self.params = params
        self.vocab_size = param
        self.n_layers = params.
```

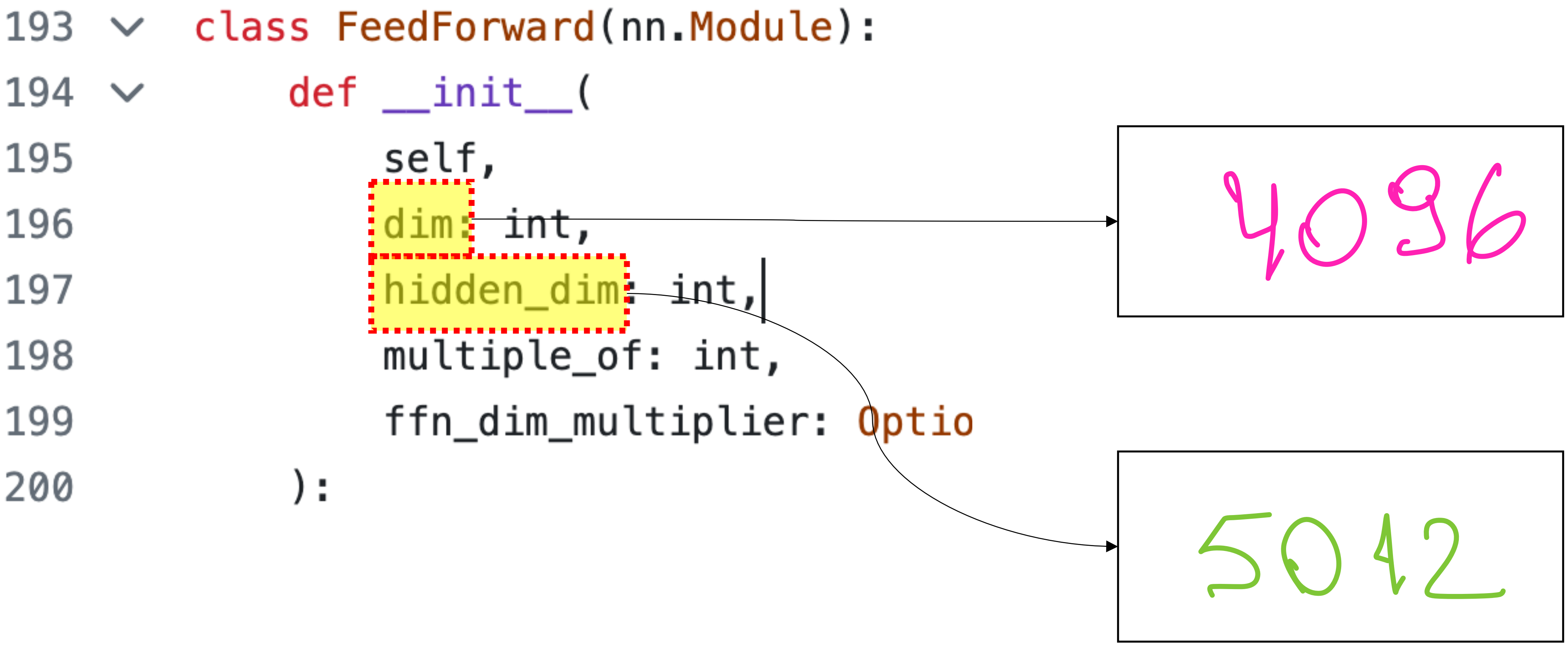
AI by Hand ✍️ 2024 © Tom Yeh

Self Attention



```
class Attention(nn.Module):
    def __init__(self, args: ModelArgs):
        super().__init__()
        self.n_kv_heads = args.n_heads if args.n_k
        model_parallel_size = fs_init.get_model_pa
        self.n_local_heads = args.n_heads // model
        self.n_local_kv_heads = self.n_kv_heads //
        self.n_rep = self.n_local_heads // self.n_
        self.head_dim = args.dim // args.n_heads
```

Feed Forward



```
class FeedForward(nn.Module):
    def __init__(
        self,
        dim: int,
        hidden_dim: int,
        multiple_of: int,
        ffn_dim_multiplier: Optio
    ):
```

AI by Hand ✍️ 2024 © Tom Yeh

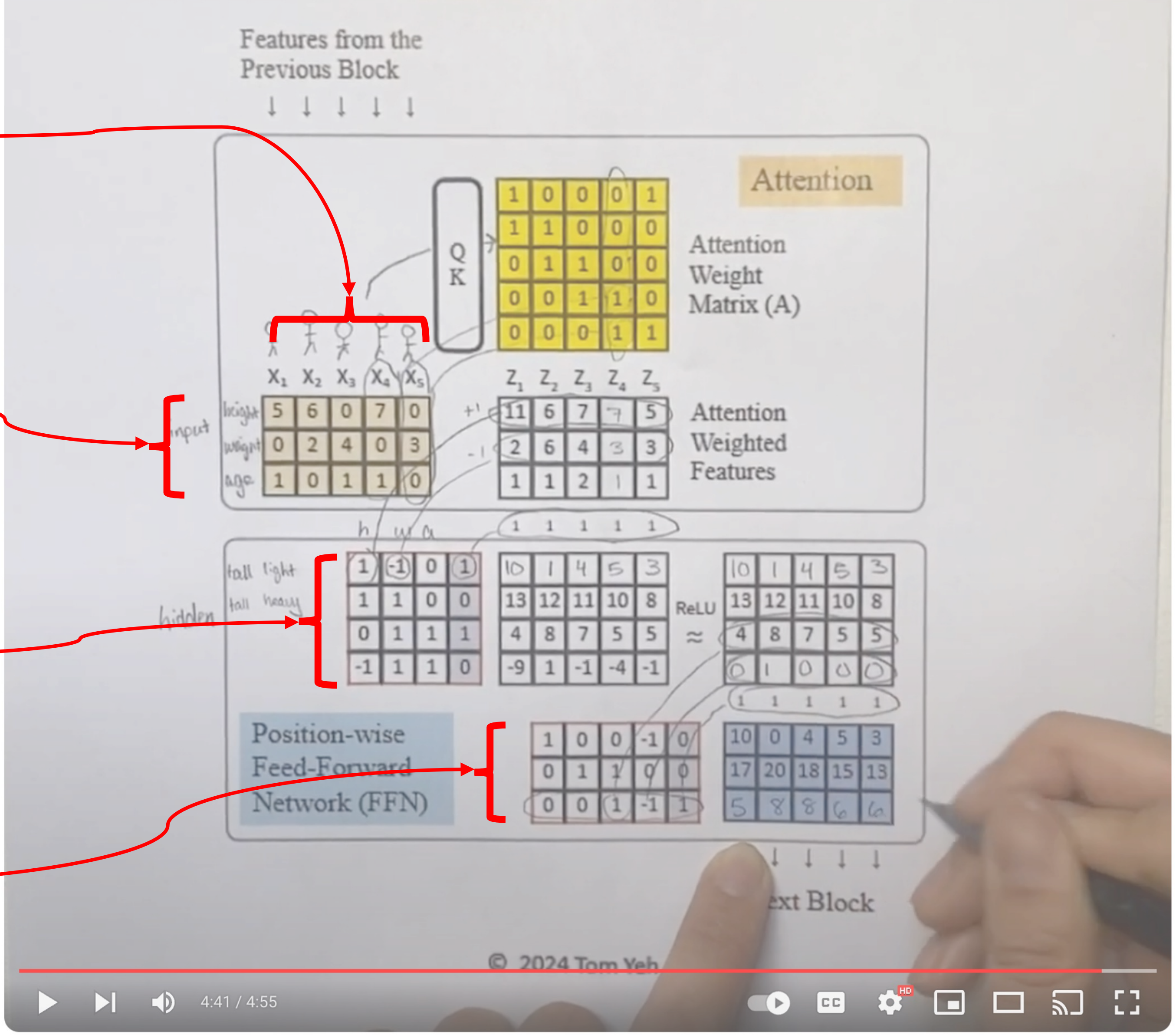
Transformer Block

8K

4096

5012

4096



Features from the Previous Block

Attention

	X₁	X₂	X₃	X₄	X₅
height	5	6	0	7	0
weight	0	2	4	0	3
age	1	0	1	1	0

input

Q K

Attention Weight Matrix (A)

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

Attention Weighted Features

	Z₁	Z₂	Z₃	Z₄	Z₅
+1	11	6	7	7	5
-1	2	6	4	3	3
	1	1	2	1	1
	1	1	1	1	1

hidden

	h	w	a	
tall light	1	-1	0	1
tall heavy	1	1	0	0
	0	1	1	1
	-1	1	1	0

10	1	4	5	3
13	12	11	10	8
4	8	7	5	5
-9	1	-1	-4	-1

ReLU ≈

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

Position-wise Feed-Forward Network (FFN)

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

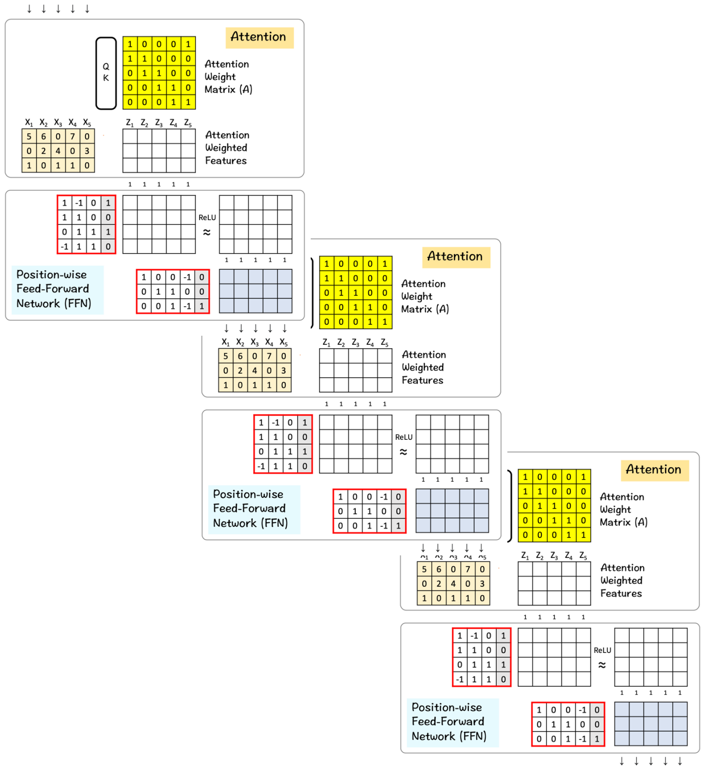
ext Block

© 2024 Tom Yeh

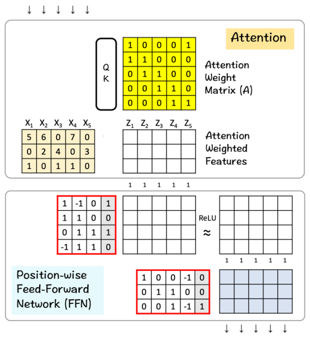
S2.E1 - Transformer - AI by Hand ✍️ with Anna

AI by Hand ✍️ 2024 © Tom Yeh

Layers


Attention
Q
K
Attention Weight Matrix (A)
X_1 X_2 X_3 X_4 X_5
Z_1 Z_2 Z_3 Z_4 Z_5
Attention Weighted Features
ReLU
≈
Position-wise Feed-Forward Network (FFN)

......


Attention
Q
K
Attention Weight Matrix (A)
X_1 X_2 X_3 X_4 X_5
Z_1 Z_2 Z_3 Z_4 Z_5
Attention Weighted Features
ReLU
≈
Position-wise Feed-Forward Network (FFN)

$$\begin{bmatrix}1&0&0&0&1\\1&1&0&0&0\\0&1&1&0&0\\0&0&1&1&0\\0&0&0&1&1\end{bmatrix}$$

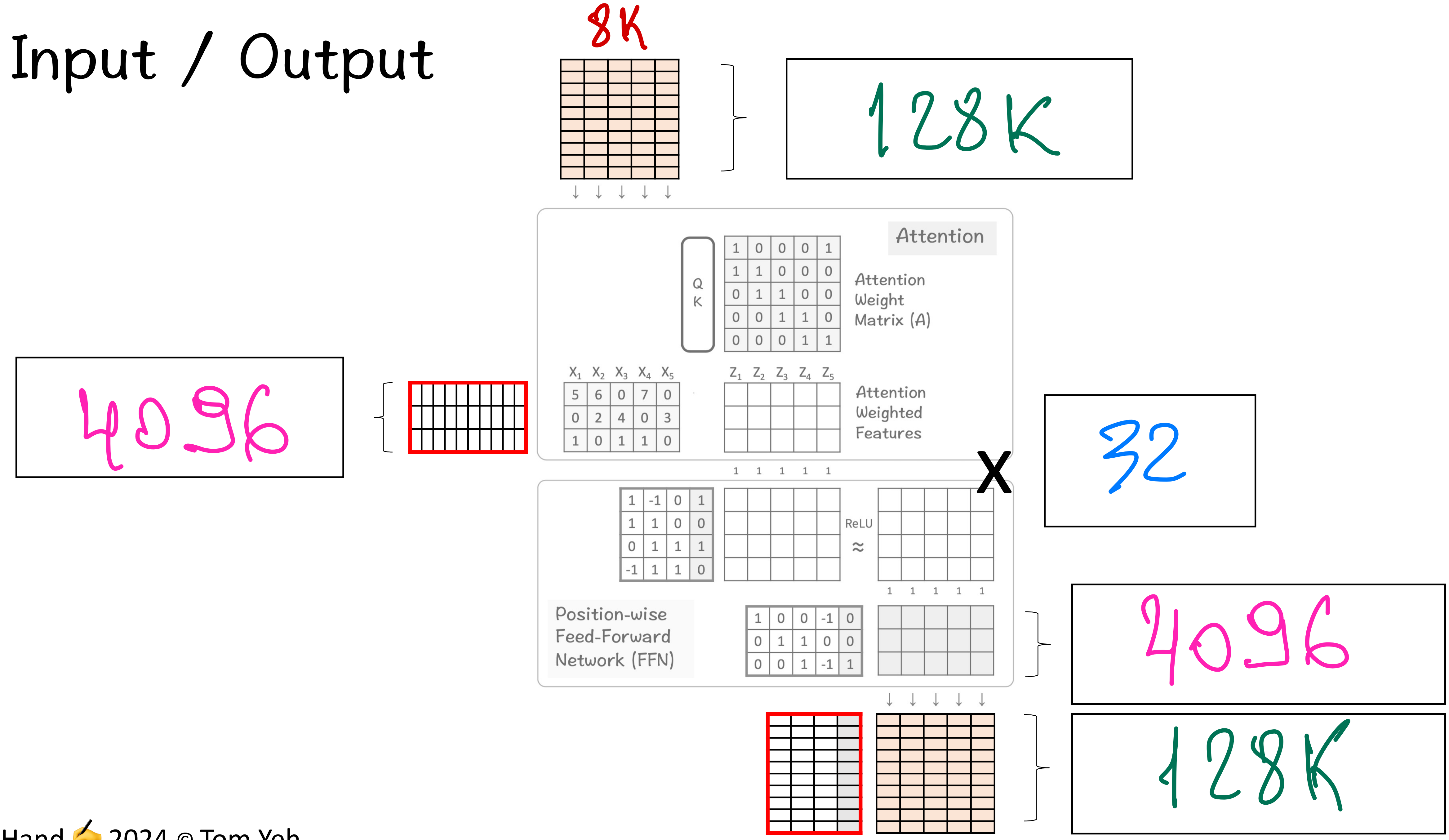
$$\begin{array}{ccccc}X_1&X_2&X_3&X_4&X_5\\5&6&0&7&0\\0&2&4&0&3\\1&0&1&1&0\end{array}$$

$$\begin{bmatrix}1&-1&0&1\\1&1&0&0\\0&1&1&1\\-1&1&1&0\end{bmatrix}\qquad\begin{bmatrix}1&0&0&-1&0\\0&1&1&0&0\\0&0&1&-1&1\end{bmatrix}$$

x 32

AI by Hand ✍️ 2024 © Tom Yeh

Input / Output



8K
128K
Attention
Q
K
Attention Weight Matrix (A)
X_1 X_2 X_3 X_4 X_5
Z_1 Z_2 Z_3 Z_4 Z_5
Attention Weighted Features
4096
32
X
1 1 1 1 1
ReLU
≈
1 1 1 1 1
Position-wise Feed-Forward Network (FFN)
4096
128K

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

X_1	X_2	X_3	X_4	X_5
5	6	0	7	0
0	2	4	0	3
1	0	1	1	0

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

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