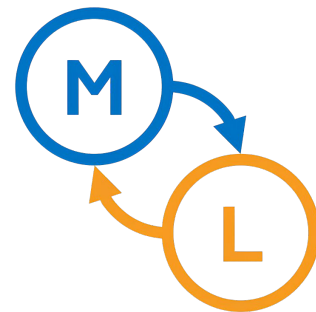


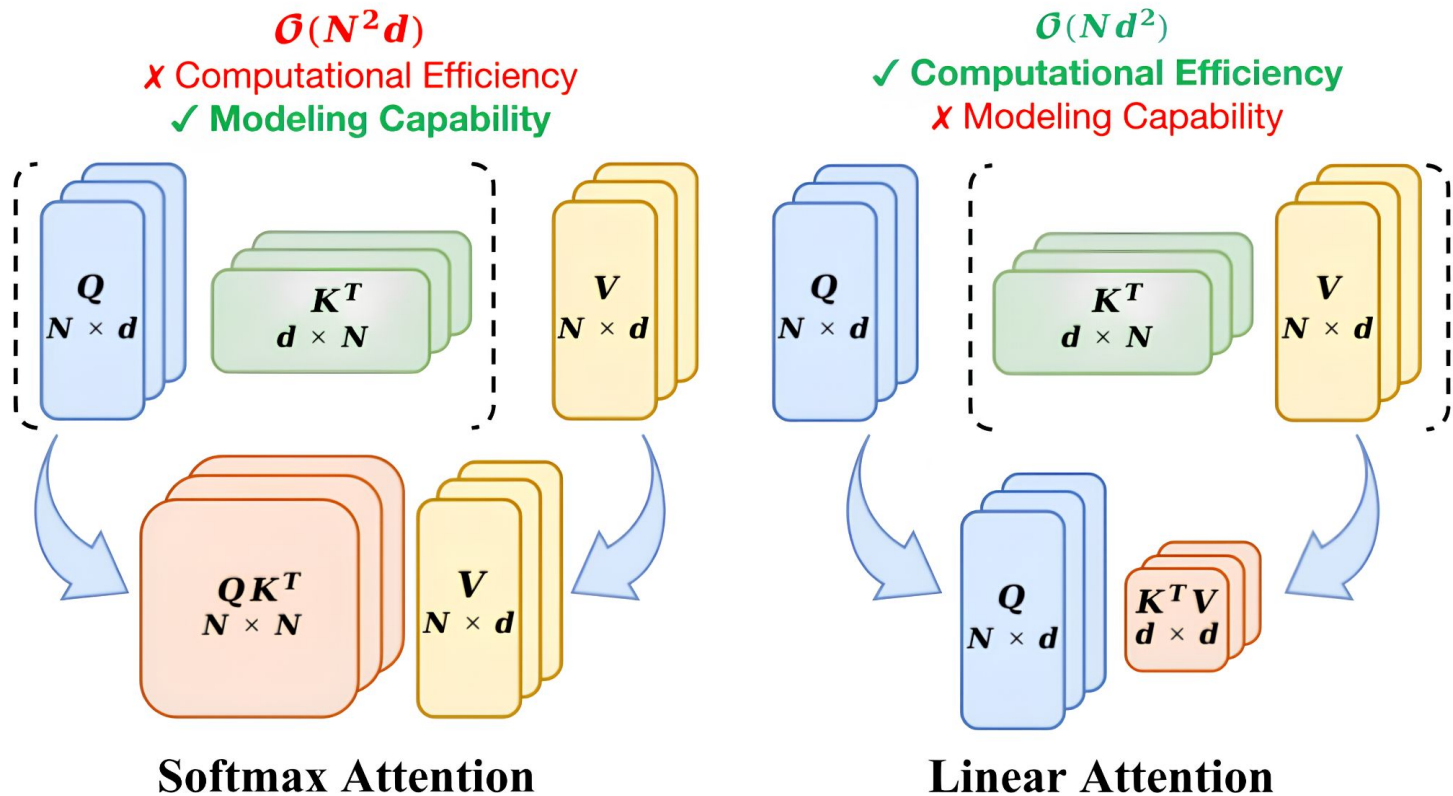
Mamba 2 - SSM and Attention mechanism

An overview by
Laetin A., Romanov M.
Markov Lab



Recap

Linear Attention



SSM

State → Output

$$x'(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t) + Du(t)$$

Parameters

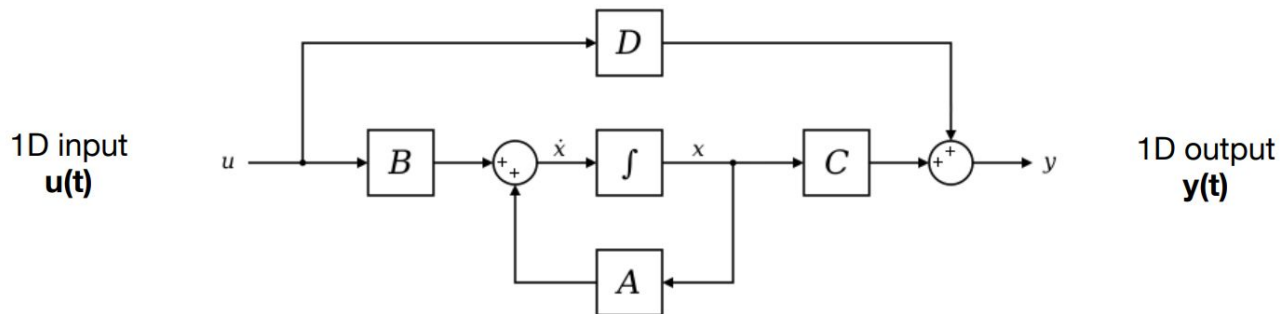
$$A \in \mathbb{R}^{N \times N}$$

$$B \in \mathbb{R}^{N \times 1}$$

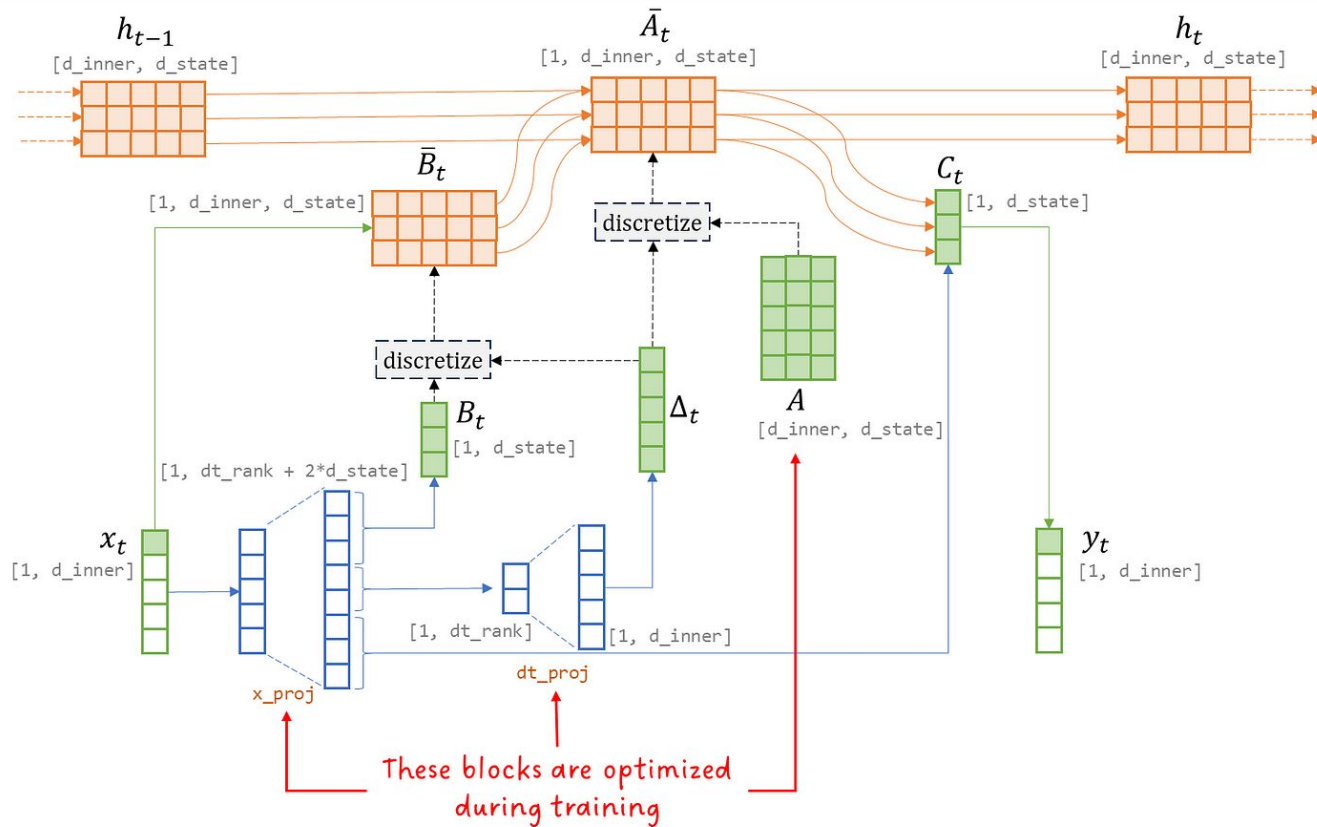
$$C \in \mathbb{R}^{1 \times N}$$

$$D \in \mathbb{R}^{1 \times 1}$$

Function-to-function map $u(t) \mapsto y(t)$



Mamba 1



The model

New notation

$$h_t = A_t h_t + B_t x_t$$

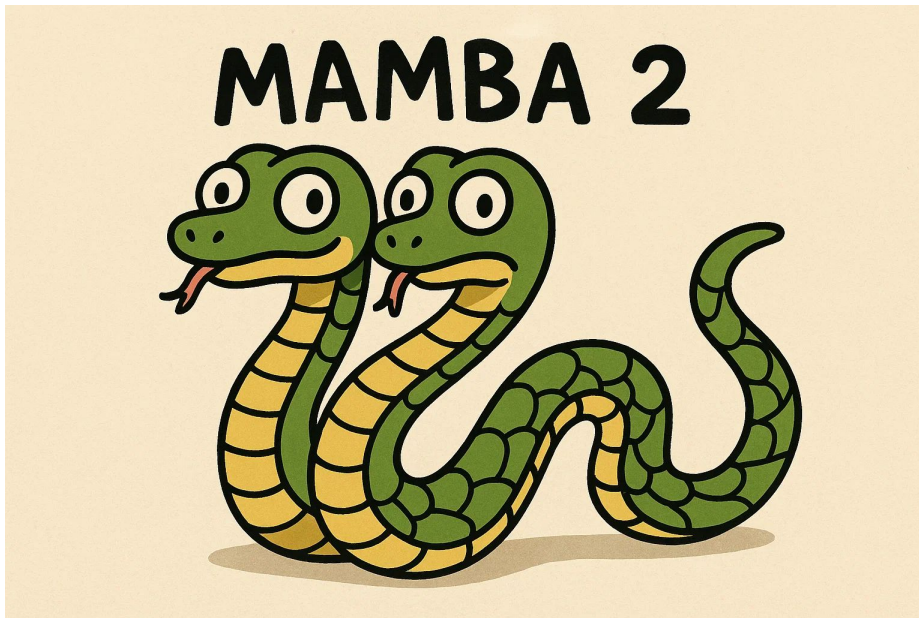
$$y_t = C_t^\top h_t$$

SSM defines a map from $x \in R^T \rightarrow y \in R^T$, where x_t and y_t are scalars, and the hidden state h_t is an N-dimensional vector, where N is an independent hyperparameter called the state size.

A selective state space model allows the (A, B, C) SSM parameters to vary across time: $A \in R^{(T, N, N)}$; $B, C \in R^{(T, N)}$.

Why mamba 2?

- SSMS are disjoint from attention
- Mamba-1 training is less efficient than attention



Contribution

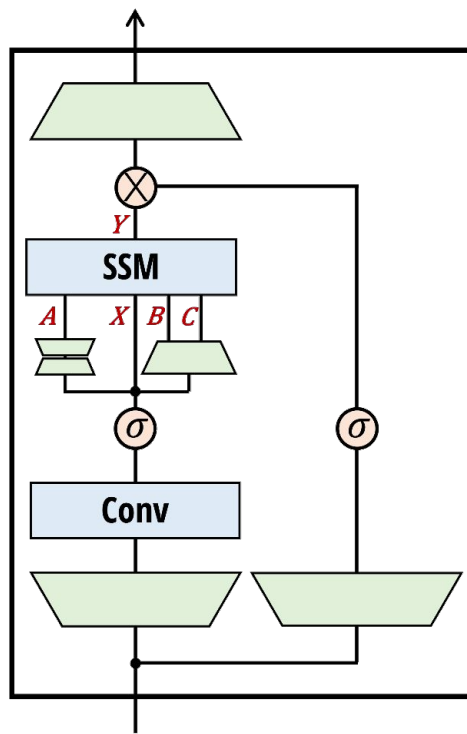
- The SSD model refers to a specific standalone layer, like attention or an SSM, that can be incorporated into deep neural networks
- The SSD algorithm is an algorithm for computing SSD layers much more efficiently than previous SSMs

Modifications (Mamba-1 vs Mamba-2)

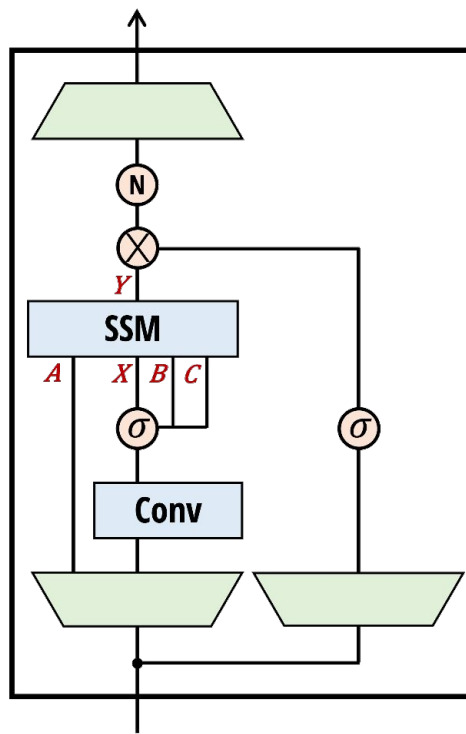
- A_t is scalar vs diagonal
- Multi-head vs different head for each channel

$$Y^{(T,P)} = \text{SSM}(A^{(T,\dots)}, B^{(T,N)}, C^{(T,N)})(X^{(T,P)})$$

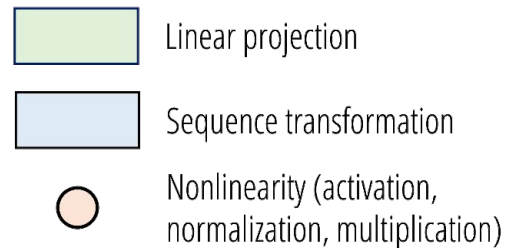
Mamba 2 architecture



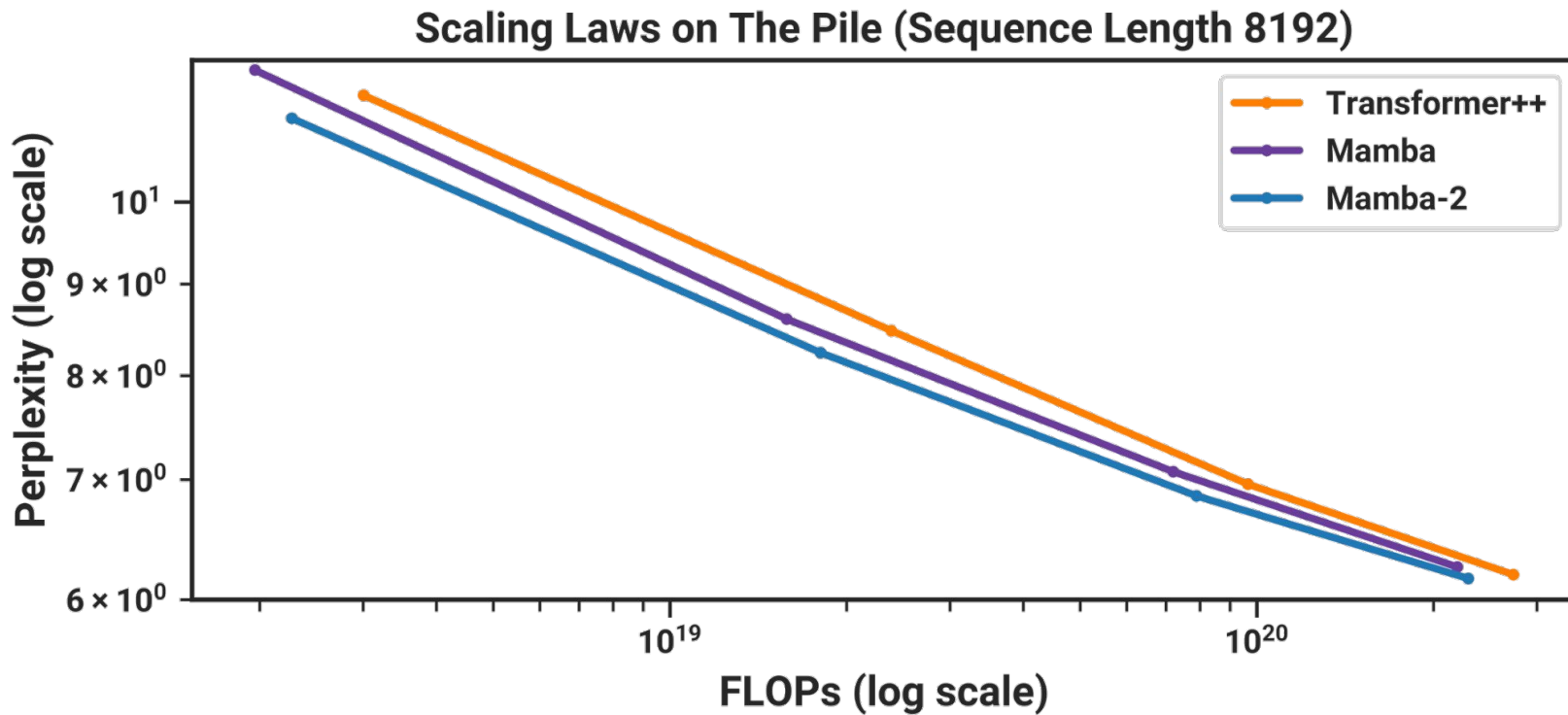
Sequential Mamba Block



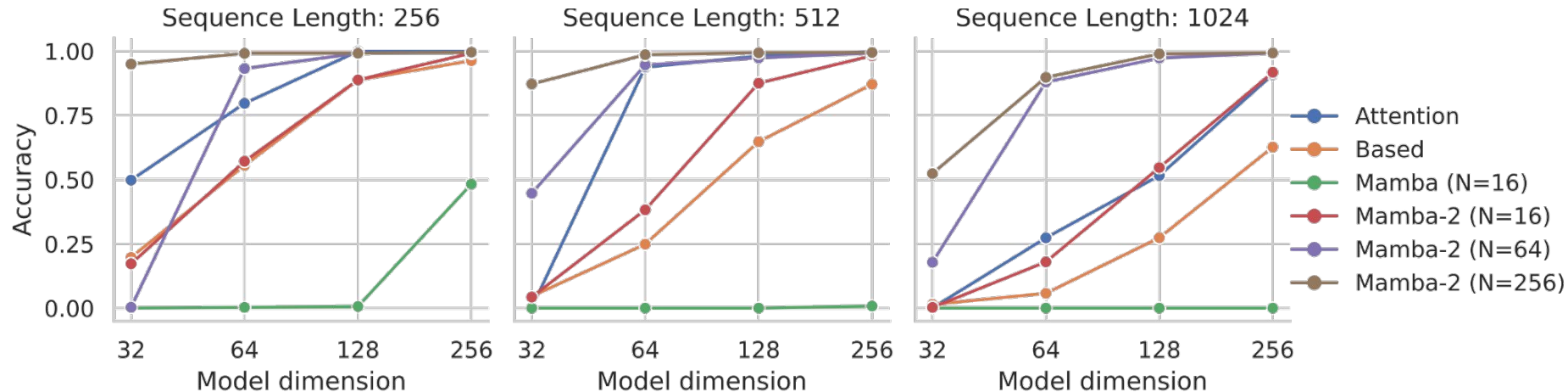
Parallel Mamba Block



Metrics and conclusion



Metrics and conclusion



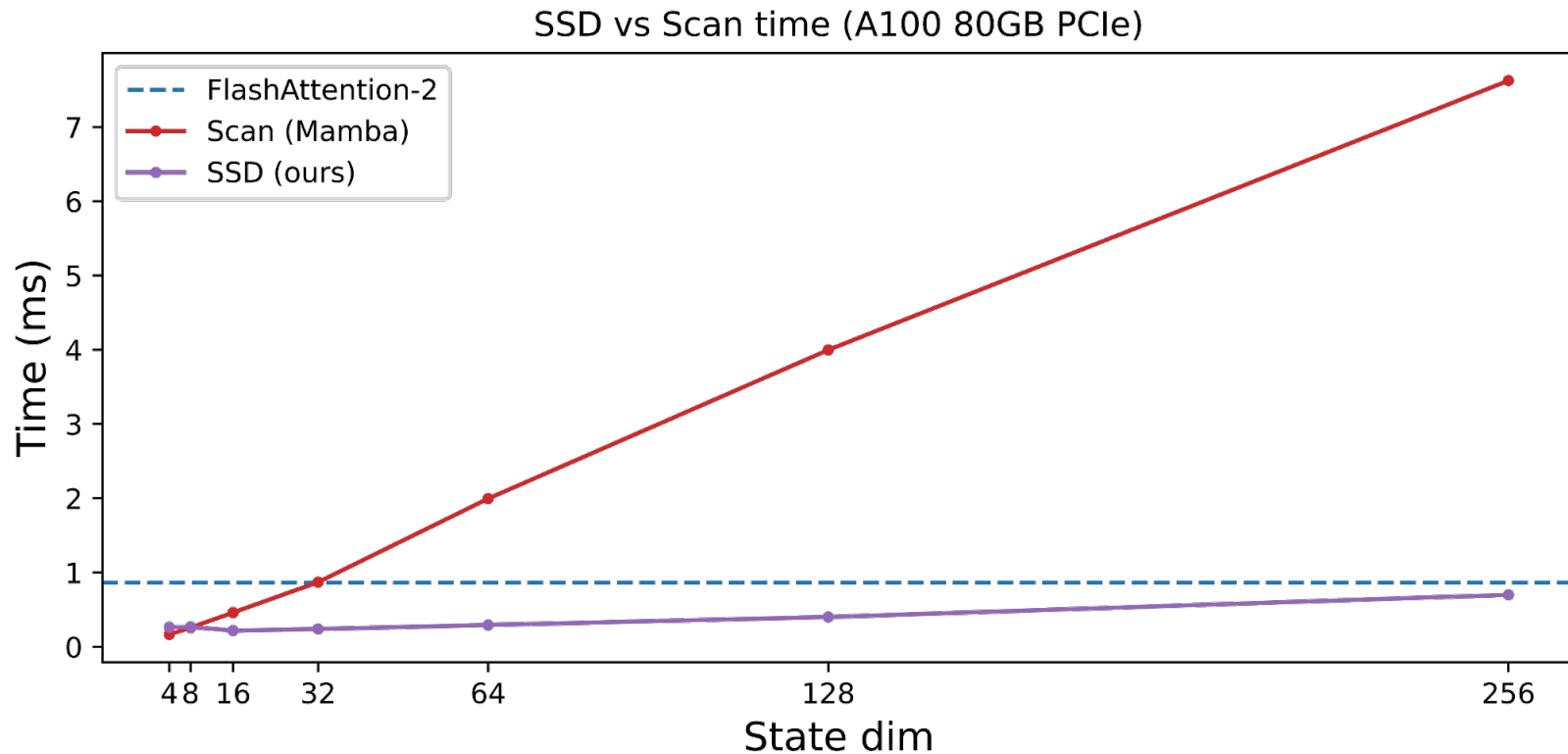
Metrics and conclusion

MODEL	TOKEN.	PILE PPL ↓	LAMBADA PPL ↓	LAMBADA ACC ↑	HELLASWAG ACC ↑	PIQA ACC ↑	ARC-E ACC ↑	ARC-C ACC ↑	WINOGRANDE ACC ↑	OPENBOOKQA ACC ↑	AVERAGE ACC ↑
Pythia-1B	NeoX	7.82	7.92	56.1	47.2	70.7	57.0	27.1	53.5	31.4	49.0
Mamba-790M	NeoX	<u>7.33</u>	<u>6.02</u>	62.7	55.1	72.1	61.2	29.5	<u>56.1</u>	<u>34.2</u>	<u>53.0</u>
Mamba-2-780M	NeoX	7.26	5.86	<u>61.7</u>	<u>54.9</u>	<u>72.0</u>	<u>61.0</u>	<u>28.5</u>	60.2	36.2	53.5
Hybrid H3-1.3B	GPT2	—	11.25	49.6	52.6	71.3	59.2	28.1	56.9	34.4	50.3
Pythia-1.4B	NeoX	7.51	6.08	61.7	52.1	71.0	60.5	28.5	57.2	30.8	51.7
RWKV4-1.5B	NeoX	7.70	7.04	56.4	52.5	72.4	60.5	29.4	54.6	34.0	51.4
Mamba-1.4B	NeoX	<u>6.80</u>	<u>5.04</u>	<u>65.0</u>	<u>59.1</u>	74.2	65.5	<u>32.8</u>	61.5	<u>36.4</u>	56.4
Mamba-2-1.3B	NeoX	6.66	5.02	65.7	59.9	<u>73.2</u>	<u>64.3</u>	33.3	<u>60.9</u>	37.8	56.4
Hybrid H3-2.7B	GPT2	—	7.92	55.7	59.7	73.3	65.6	32.3	61.4	33.6	54.5
Pythia-2.8B	NeoX	6.73	5.04	64.7	59.3	74.0	64.1	32.9	59.7	35.2	55.7
RWKV4-3B	NeoX	7.00	5.24	63.9	59.6	73.7	67.8	33.1	59.6	37.0	56.4
Mamba-2.8B	NeoX	<u>6.22</u>	<u>4.23</u>	<u>69.2</u>	<u>66.1</u>	<u>75.2</u>	69.7	<u>36.3</u>	<u>63.5</u>	39.6	<u>59.9</u>
Mamba-2-2.7B	NeoX	6.09	4.10	69.7	66.6	76.4	<u>69.6</u>	36.4	64.0	<u>38.8</u>	60.2

Metrics and conclusion

MODEL	TOKEN.	PILE PPL ↓	LAMBADA PPL ↓	LAMBADA ACC ↑	HELLASWAG ACC ↑	PIQA ACC ↑	ARC-E ACC ↑	ARC-C ACC ↑	WINOGRANDE ACC ↑	OPENBOOKQA ACC ↑	AVERAGE ACC ↑
Transformer++	NeoX	6.13	3.99	70.3	66.4	75.2	67.7	37.8	63.9	40.4	60.2
Mamba-2	NeoX	6.09	4.10	69.7	66.6	76.4	69.6	36.4	64.0	38.8	60.2
Mamba-2-Attention	NeoX	5.95	3.85	71.1	67.8	75.8	69.9	37.8	65.3	39.0	61.0

Metrics and conclusion



The Theory (SSD)

Attention view (duality and comparison with linear att)

$$L = \begin{bmatrix} 1 & & & & \\ a_1 & 1 & & & \\ a_2 a_1 & a_2 & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ a_{T-1} \dots a_1 & a_{T-1} \dots a_2 & \dots & a_{T-1} & 1 \end{bmatrix}.$$

$$M = L \circ CB^\top \in \mathbb{R}^{(T, T)}$$

Attention view (why it works so)

$$h_0 = B_0 x_0,$$

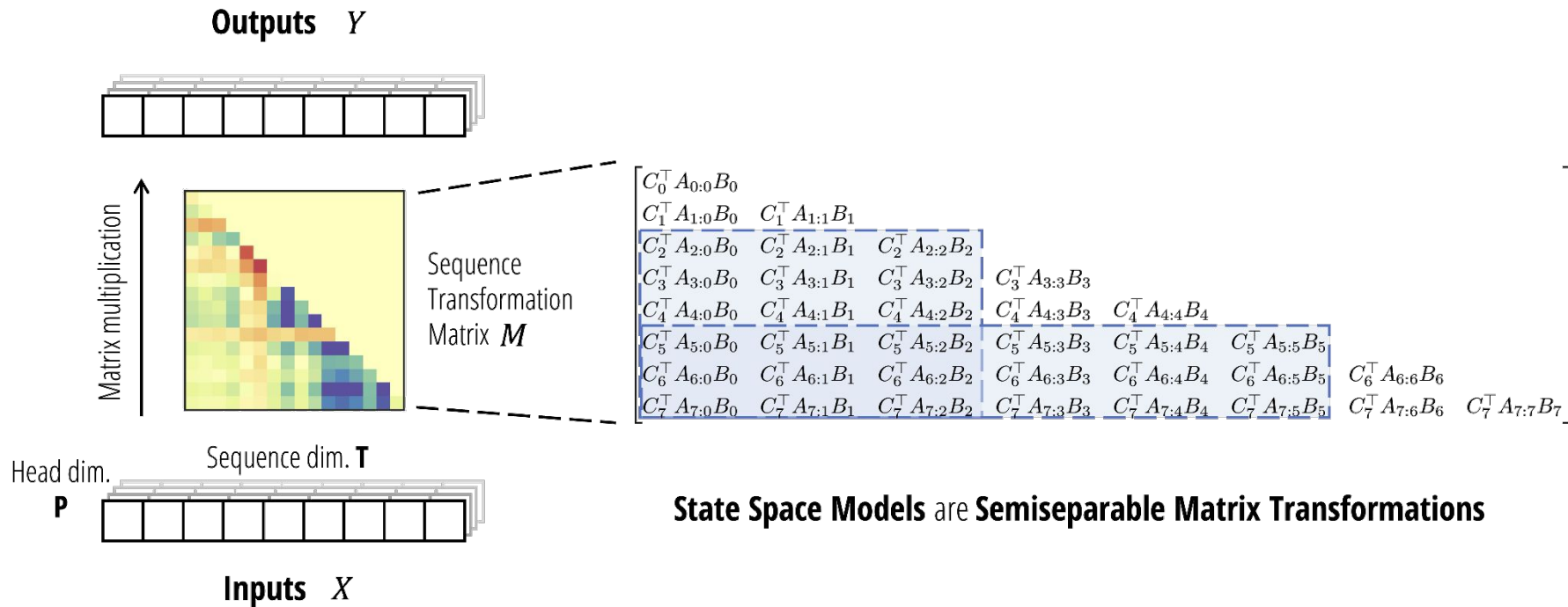
$$h_1 = B_1 x_1 + a_1 (B_0 x_0),$$

$$h_2 = B_2 x_2 + a_2 B_1 x_1 + a_1 a_2 B_0 x_0 + \dots$$

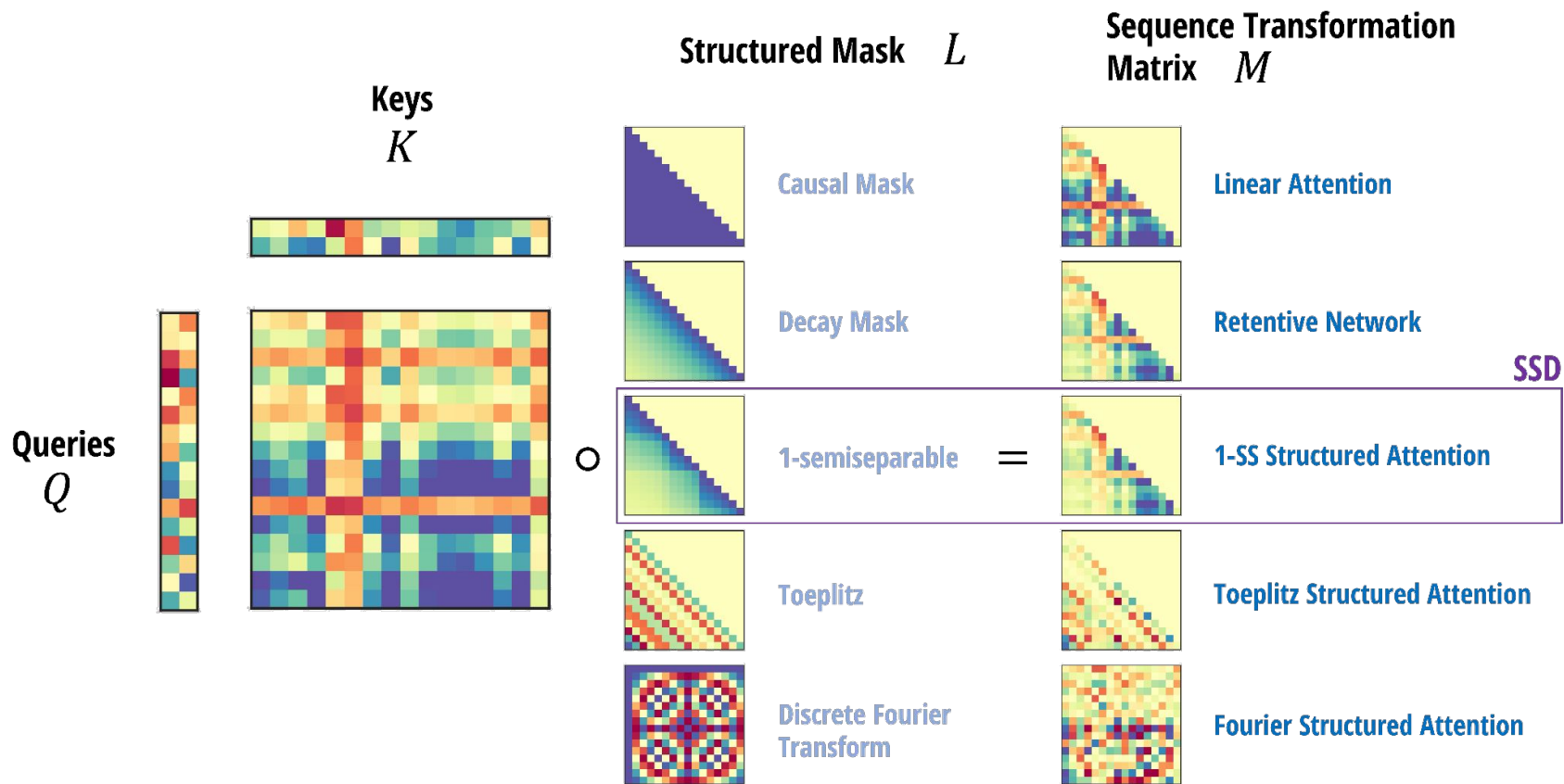
$$y_t = C_t h_t$$

$\Rightarrow y = Mx$, where M is an “attention-like” matrix.

Attention view



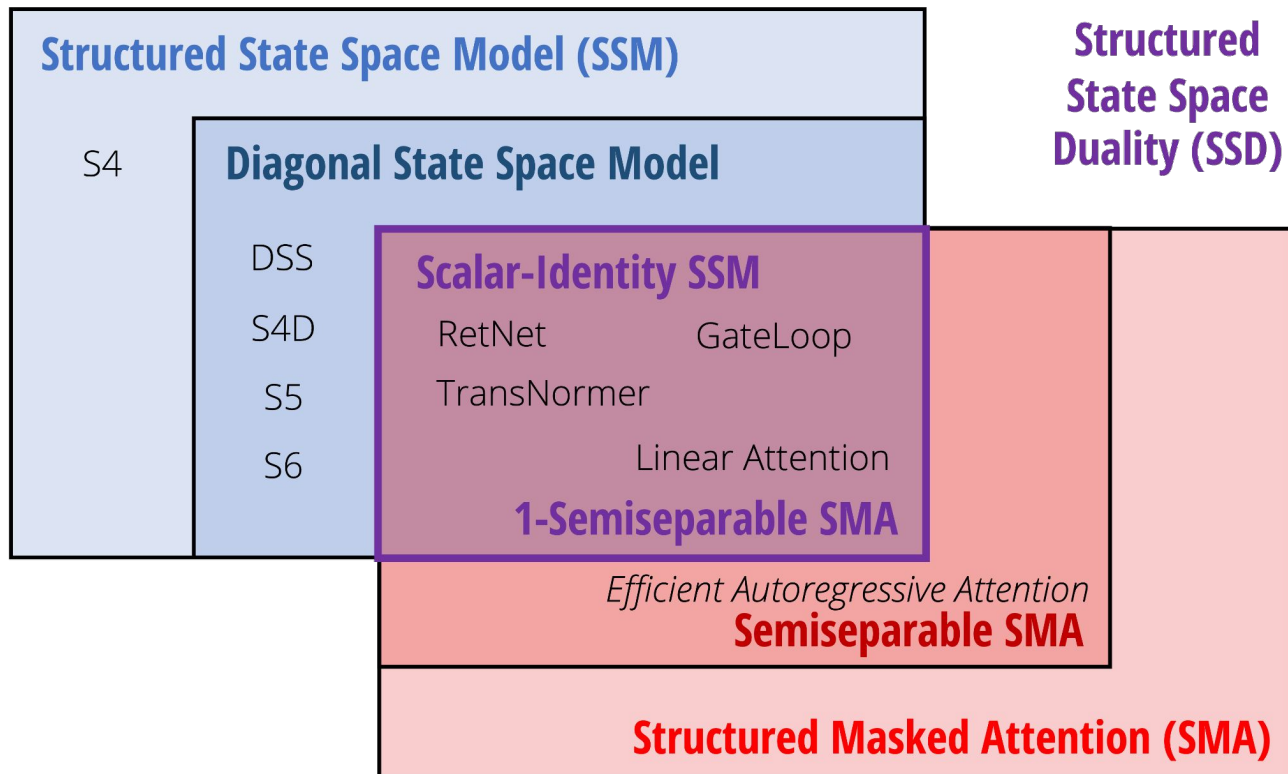
SSD summary



SSD vs SSM vs Attention

	Attention	SSM	SSD
State size	T	N	N
Training FLOPs	T^2N	TN^2	TN^2
Inference FLOPs	TN	N^2	N^2
(Naive) memory	T^2	TN^2	TN
Matrix multiplications?	✓	✗	✓

SSD conclusion



The Algorithm

Mamba-2 vs Mamba-1

- Mamba-1 bottleneck - parallel associative scan (no tensor cores)
- H100: 989 TFLOPS of BF16 matmul vs 67 TFLOPS of FP32 arithmetics
- Goal is to leverage matmuls

From Recurrence to Matrix Multiplication

$C_0^\top A_{0:0} B_0$ $C_1^\top A_{1:0} B_0 \quad C_1^\top A_{1:1} B_1$ $C_2^\top A_{2:0} B_0 \quad C_2^\top A_{2:1} B_1 \quad C_2^\top A_{2:2} B_2$		
$\begin{bmatrix} C_3^\top A_{3:2} \\ C_4^\top A_{4:2} \\ C_5^\top A_{5:2} \end{bmatrix}$ $A_{2:2}$ $\begin{bmatrix} B_0^\top A_{2:0} \\ B_1^\top A_{2:1} \\ B_2^\top A_{2:2} \end{bmatrix}^\top$	$C_3^\top A_{3:3} B_3$ $C_4^\top A_{4:3} B_3 \quad C_4^\top A_{4:4} B_4$ $C_5^\top A_{5:3} B_3 \quad C_5^\top A_{5:4} B_4 \quad C_5^\top A_{5:5} B_5$	
$\begin{bmatrix} C_6^\top A_{6:5} \\ C_7^\top A_{7:5} \\ C_8^\top A_{8:5} \end{bmatrix}$ $A_{5:5}$ $\begin{bmatrix} B_0^\top A_{2:0} \\ B_1^\top A_{2:1} \\ B_2^\top A_{2:2} \end{bmatrix}^\top$	$\begin{bmatrix} C_6^\top A_{6:5} \\ C_7^\top A_{7:5} \\ C_8^\top A_{8:5} \end{bmatrix}$ $A_{5:5}$ $\begin{bmatrix} B_3^\top A_{5:3} \\ B_4^\top A_{5:4} \\ B_5^\top A_{5:5} \end{bmatrix}^\top$	$C_6^\top A_{6:6} B_6$ $C_7^\top A_{7:6} B_6 \quad C_7^\top A_{7:7} B_7$ $C_8^\top A_{8:6} B_6 \quad C_8^\top A_{8:7} B_7 \quad C_8^\top A_{8:8} B_8$

Semiseparable Matrix M Block Decomposition

- Diagonal Block: Input $\rightarrow$ Output
- Low-Rank Block: Input $\rightarrow$ State
- Low-Rank Block: State $\rightarrow$ State
- Low-Rank Block: State $\rightarrow$ Output

