ETH zürich

Eric Nothum

Mamba: Linear-Time Sequence Modeling with Selective State Spaces

Albert Gu, Tri Dao

Motivation for sequence modeling

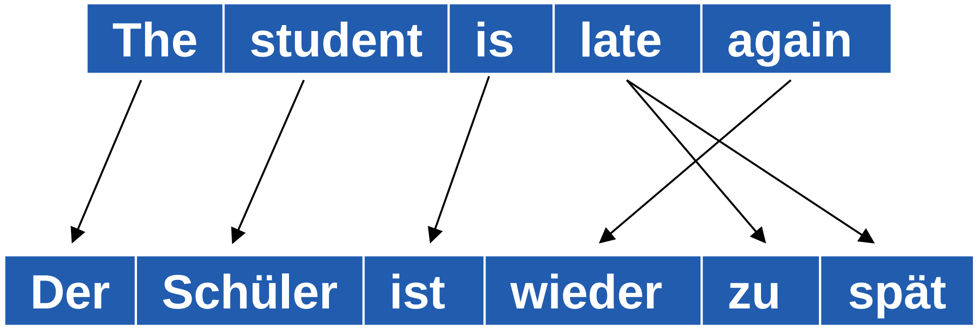
What is a sequence?

x_0	x_1	x_2	x_3	…	x_t	x_{t+1}	x_{t+2}	x_{t+3}

Example of a sequence: **Text**

The	student	is	late	again

Example of a sequence task: **Translation**



The student is late again

Der Schüler ist wieder zu spät

Motivation for sequences

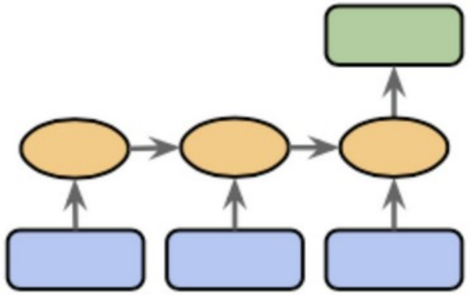
- **Videos are sequences of images**
- Tasks on videos:
  - Video generation
  - Video captioning

Sources: https://hallobloggi.de/wp-content/uploads/2017/03/HalloBloggi-Daumenkino-sw-01-15.jpg, https://openai.com/sora

Motivation for sequence modeling

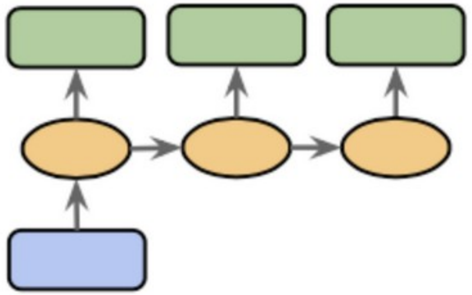
- **Audio**: speech processing and generation
- **Genomics:** process DNA sequences
- **Time series**: process data from sensors

Types of sequence tasks



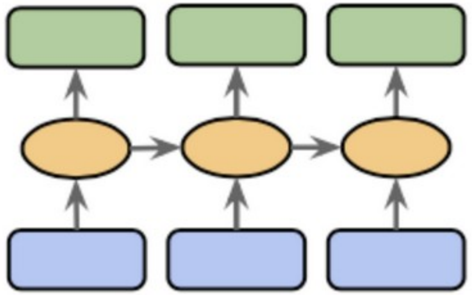
many-to-one

e.g. Sentiment classification



one-to-many

e.g. Image captioning



many-to-many

e.g. Annotate video frames

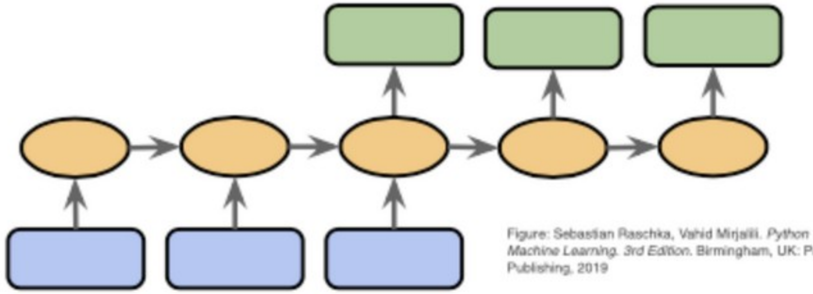


Figure: Sebastian Raschka, Vahid Mirjalili. *Python Machine Learning. 3rd Edition*. Birmingham, UK: Packt Publishing, 2019

many-to-many

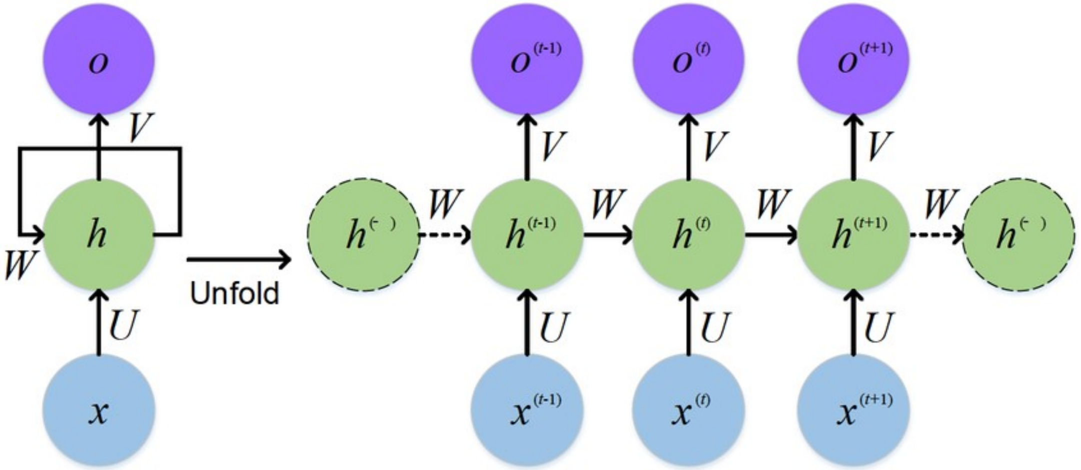
e.g. Video captioning

Source: Python Machine Learning, 3rd Ed., by Raschka, Sebastian and Mirjalili, Vahid

Related Work

RNN

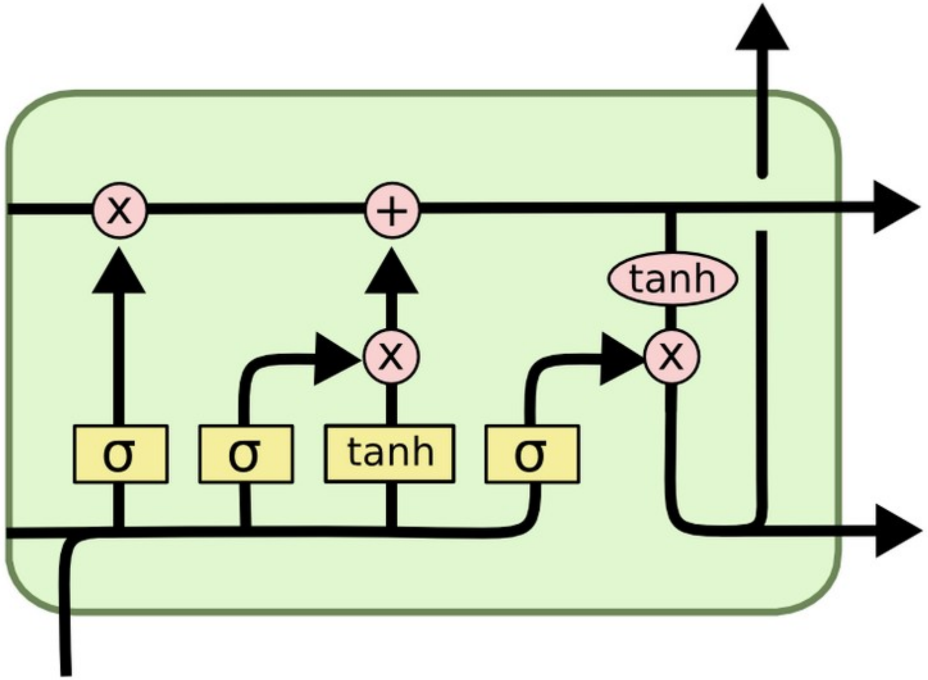
- **O(n)**
- Suffers from vanishing/exploding gradients



Source: https://www.sciencedirect.com/science/article/pii/S0065245819300506

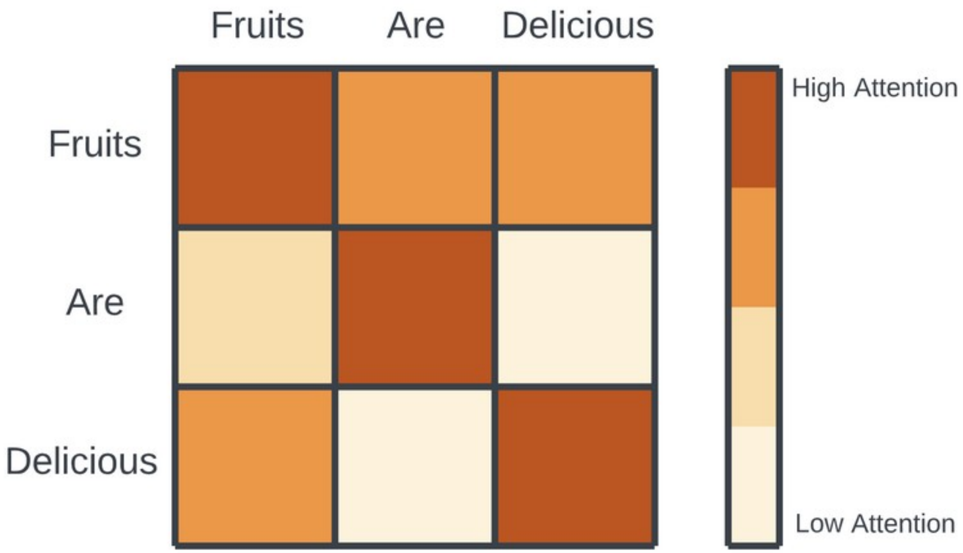
LSTM

- Keeps a long term state
- Employs gating mechanisms that allows to selectively memorize and forget information



Source: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

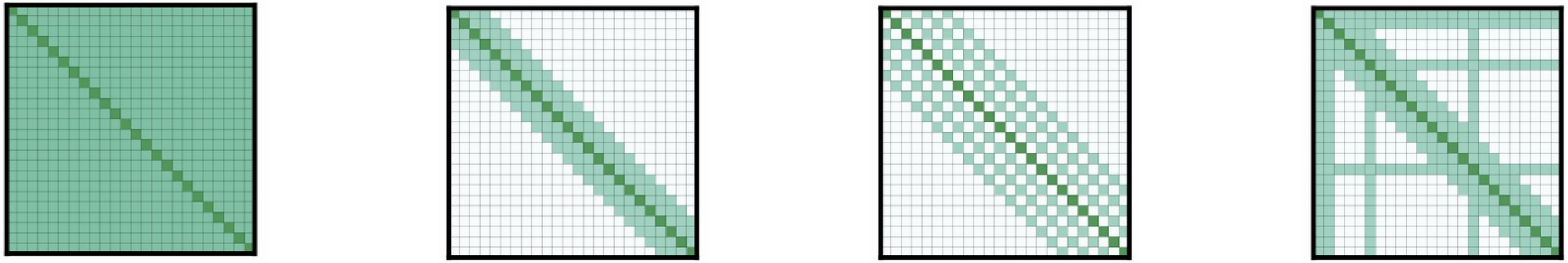
Transformer – self attention



O(n²) due to self attention

Source: https://deepgram.com/learn/visualizing-and-explaining-transformer-models-from-the-ground-up

Transformer – self attention



(a) Full n^2 attention

(b) Sliding window attention

(c) Dilated sliding window

(d) Global+sliding window

Source: Longformer: The Long-Document Transformer, by Iz Beltagy, Matthew E. Peters, Arman Cohan

Transformer: Limitations for sequences larger than the context window

The quick brown

The quick brown fox jumps over the lazy dog

State space models (SSM)

Input:

Output:

Hidden:

How to model discrete inputs like text?

Continuous **time-variant** SSM:

Continuous **time-invariant** SSM:

Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Discretized state space model

Introduced **time step**

Discretized A and B:

Discretized SSM:

- **O(n)**
- **Time invariant**
- Well suited for continuous tasks, like audio
- Not well suited for discrete tasks like text

Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Mamba

Goals

1. Build on SSMs to have linear time complexity, while
2. Matching the accuracy of transformers

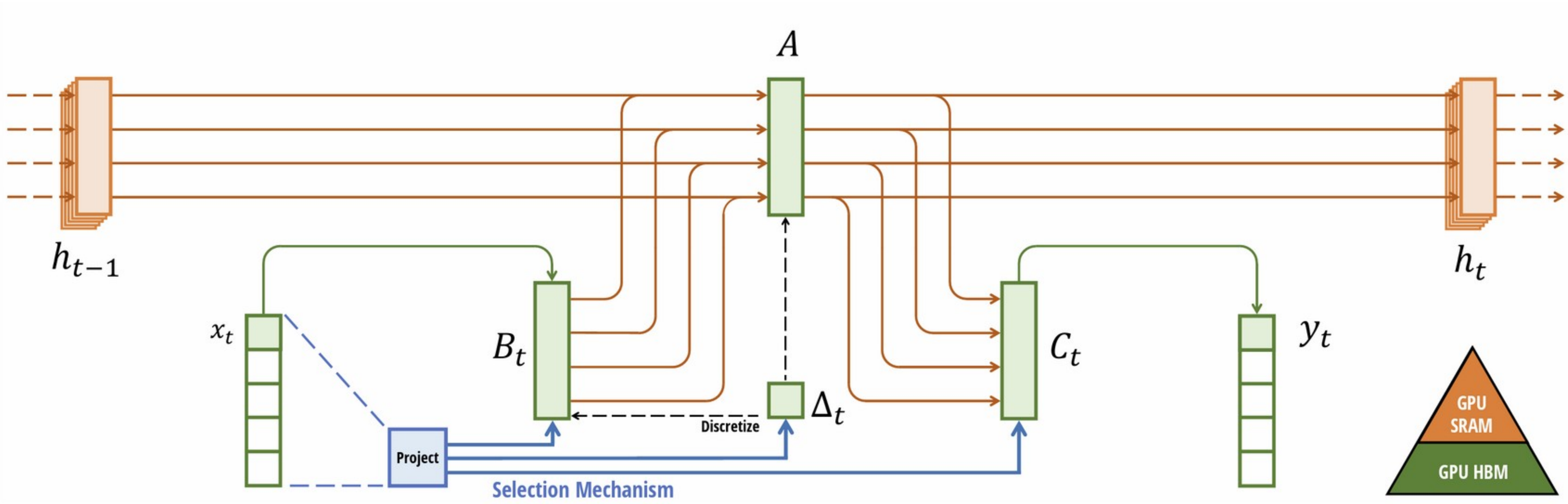
Selective state space modeling

- Selectivity:
  - **Select which inputs contribute to the hidden state**
  - Not possible with time invariant models
- Property of discretized SSMs:
  - Parameter Δ
  - Δ -> inf hidden state is reset and only current input is considered
  - Δ -> 0 hidden state is kept and current input is ignored
- Difference to previous SSMs:
  - **Δ, B, C are input dependent**

Selective SSM block

Equation of the selective SSM:

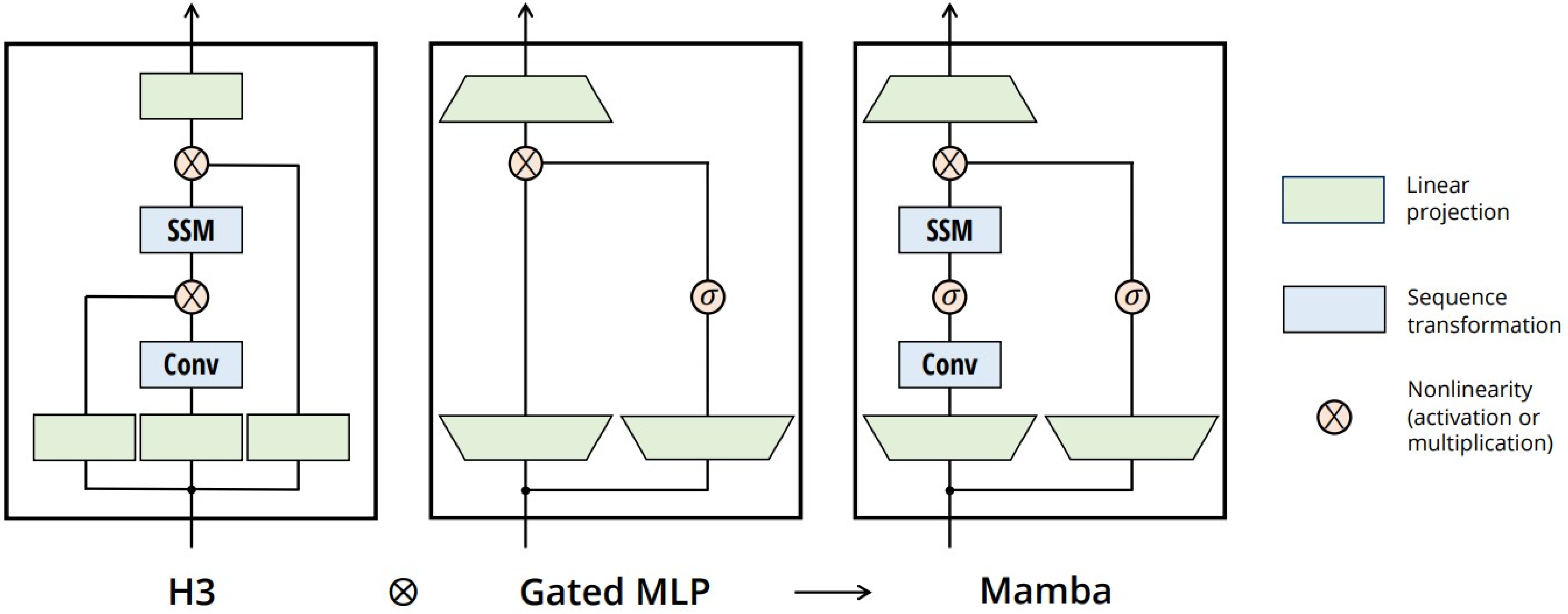
, ,



Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

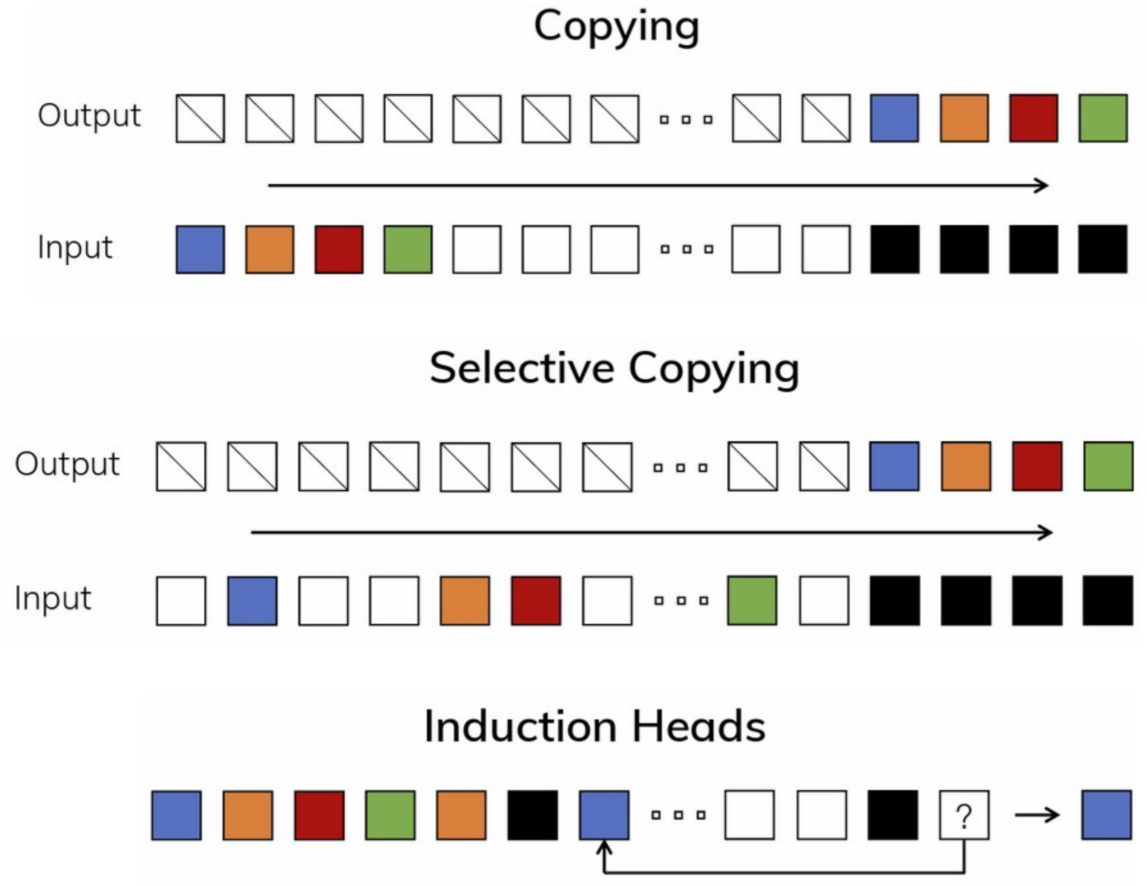
Mamba block

The selective SSM is now used in the Mamba block



Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

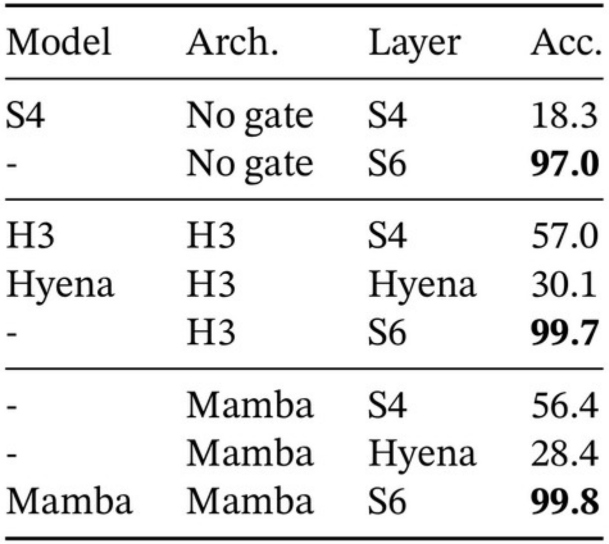
Synthetic Benchmarks



Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

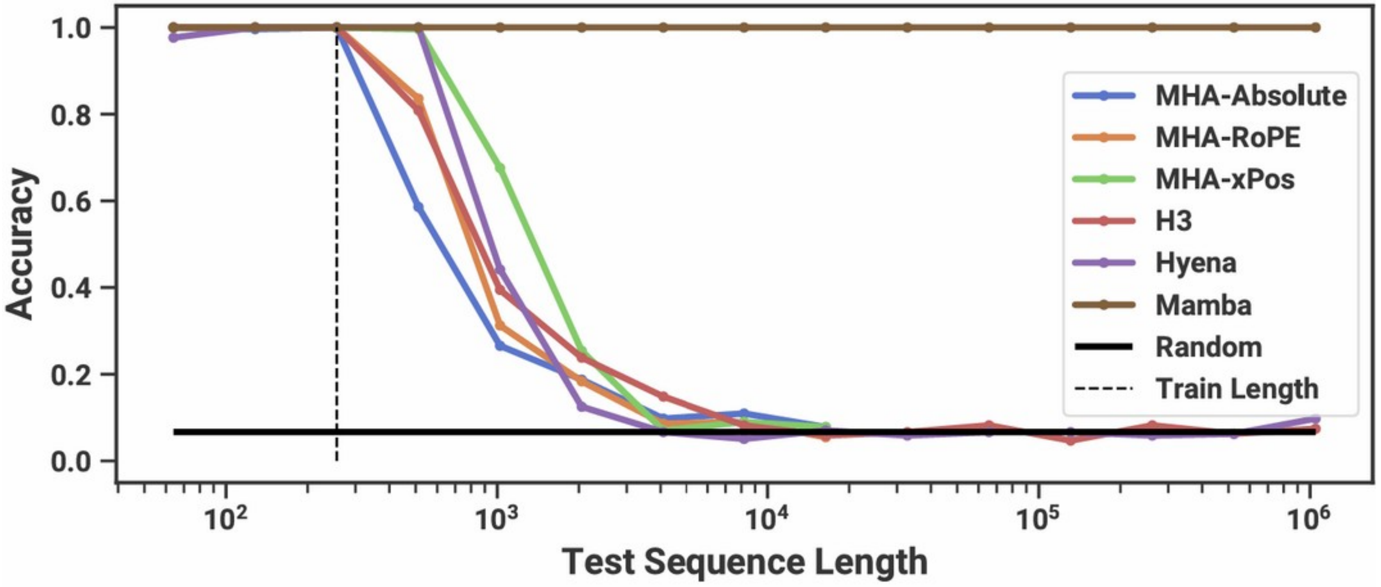
Synthetic Benchmarks

Selective copying:

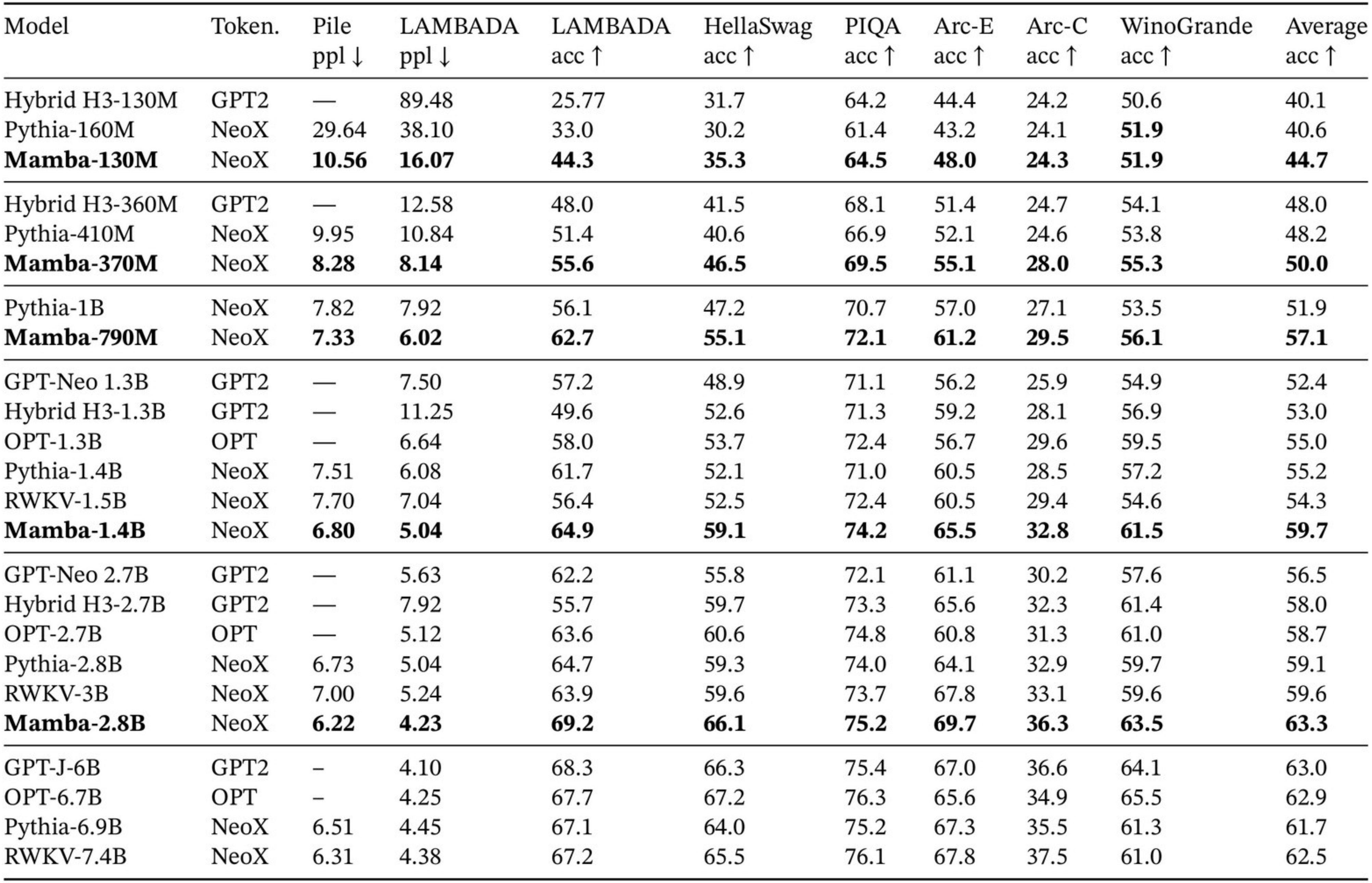


Model	Arch.	Layer	Acc.
S4	No gate	S4	18.3
-	No gate	S6	**97.0**
H3	H3	S4	57.0
Hyena	H3	Hyena	30.1
-	H3	S6	**99.7**
-	Mamba	S4	56.4
-	Mamba	Hyena	28.4
Mamba	Mamba	S6	**99.8**

Induction heads:



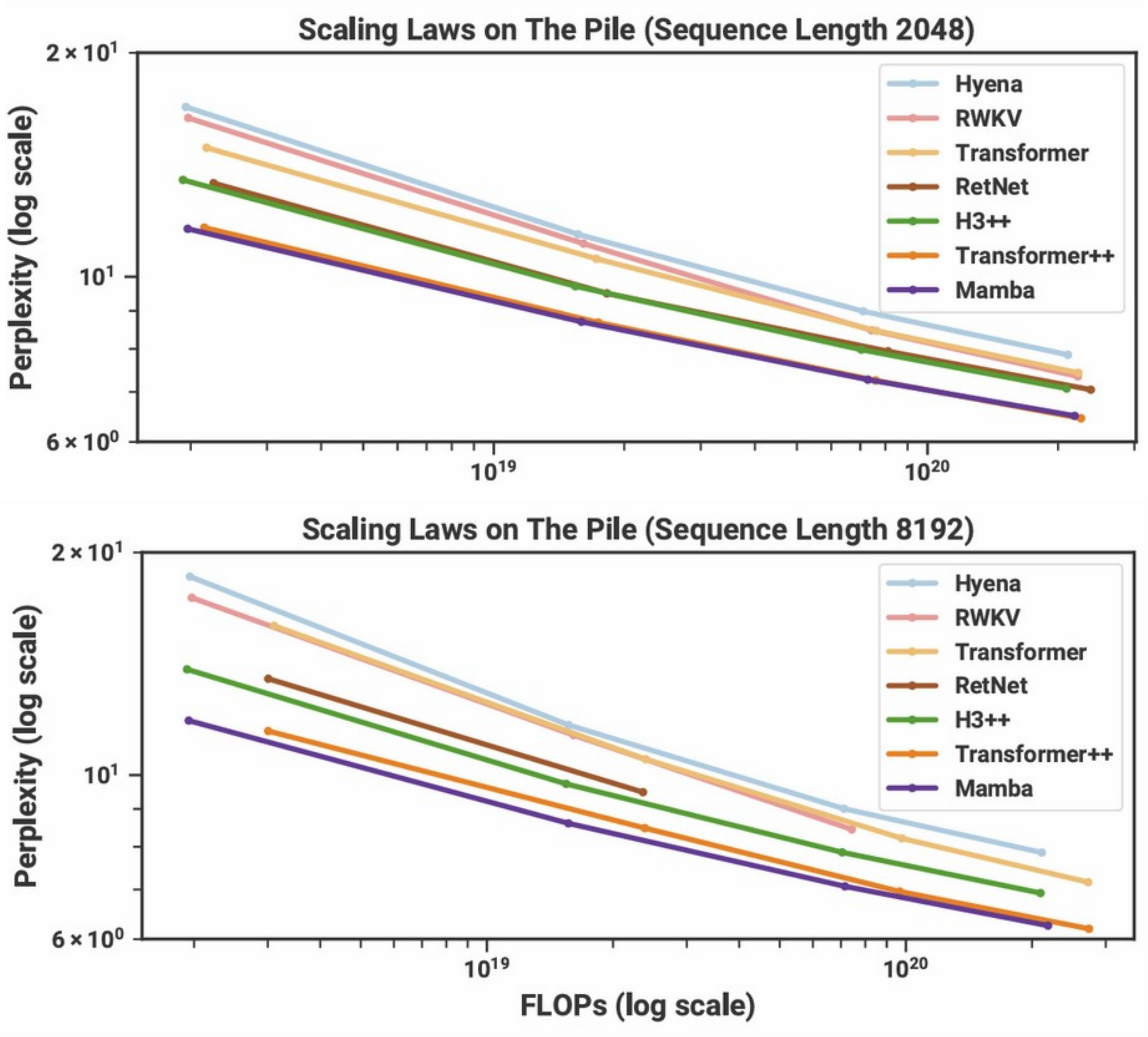
Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao



Model	Token.	Pile ppl ↓	LAMBADA ppl ↓	LAMBADA acc ↑	HellaSwag acc ↑	PIQA acc ↑	Arc-E acc ↑	Arc-C acc ↑	WinoGrande acc ↑	Average acc ↑
Hybrid H3-130M	GPT2	—	89.48	25.77	31.7	64.2	44.4	24.2	50.6	40.1
Pythia-160M	NeoX	29.64	38.10	33.0	30.2	61.4	43.2	24.1	**51.9**	40.6
Mamba-130M	NeoX	**10.56**	**16.07**	**44.3**	**35.3**	**64.5**	**48.0**	**24.3**	**51.9**	**44.7**
Hybrid H3-360M	GPT2	—	12.58	48.0	41.5	68.1	51.4	24.7	54.1	48.0
Pythia-410M	NeoX	9.95	10.84	51.4	40.6	66.9	52.1	24.6	53.8	48.2
Mamba-370M	NeoX	**8.28**	**8.14**	**55.6**	**46.5**	**69.5**	**55.1**	**28.0**	**55.3**	**50.0**
Pythia-1B	NeoX	7.82	7.92	56.1	47.2	70.7	57.0	27.1	53.5	51.9
Mamba-790M	NeoX	**7.33**	**6.02**	**62.7**	**55.1**	**72.1**	**61.2**	**29.5**	**56.1**	**57.1**
GPT-Neo 1.3B	GPT2	—	7.50	57.2	48.9	71.1	56.2	25.9	54.9	52.4
Hybrid H3-1.3B	GPT2	—	11.25	49.6	52.6	71.3	59.2	28.1	56.9	53.0
OPT-1.3B	OPT	—	6.64	58.0	53.7	72.4	56.7	29.6	59.5	55.0
Pythia-1.4B	NeoX	7.51	6.08	61.7	52.1	71.0	60.5	28.5	57.2	55.2
RWKV-1.5B	NeoX	7.70	7.04	56.4	52.5	72.4	60.5	29.4	54.6	54.3
Mamba-1.4B	NeoX	**6.80**	**5.04**	**64.9**	**59.1**	**74.2**	**65.5**	**32.8**	**61.5**	**59.7**
GPT-Neo 2.7B	GPT2	—	5.63	62.2	55.8	72.1	61.1	30.2	57.6	56.5
Hybrid H3-2.7B	GPT2	—	7.92	55.7	59.7	73.3	65.6	32.3	61.4	58.0
OPT-2.7B	OPT	—	5.12	63.6	60.6	74.8	60.8	31.3	61.0	58.7
Pythia-2.8B	NeoX	6.73	5.04	64.7	59.3	74.0	64.1	32.9	59.7	59.1
RWKV-3B	NeoX	7.00	5.24	63.9	59.6	73.7	67.8	33.1	59.6	59.6
Mamba-2.8B	NeoX	**6.22**	**4.23**	**69.2**	**66.1**	**75.2**	**69.7**	**36.3**	**63.5**	**63.3**
GPT-J-6B	GPT2	–	4.10	68.3	66.3	75.4	67.0	36.6	64.1	63.0
OPT-6.7B	OPT	–	4.25	67.7	67.2	76.3	65.6	34.9	65.5	62.9
Pythia-6.9B	NeoX	6.51	4.45	67.1	64.0	75.2	67.3	35.5	61.3	61.7
RWKV-7.4B	NeoX	6.31	4.38	67.2	65.5	76.1	67.8	37.5	61.0	62.5

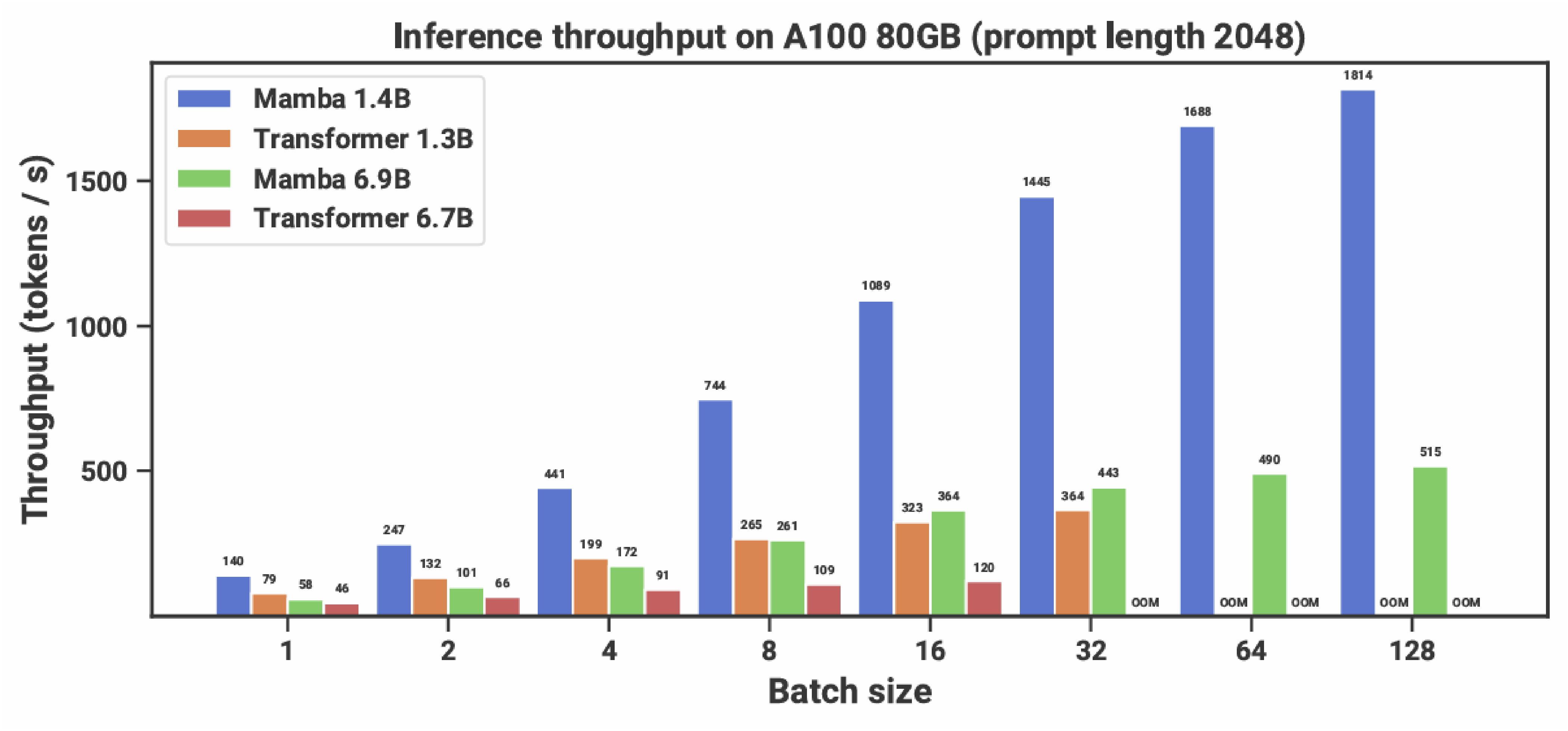
Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Language Modeling – Scaling laws



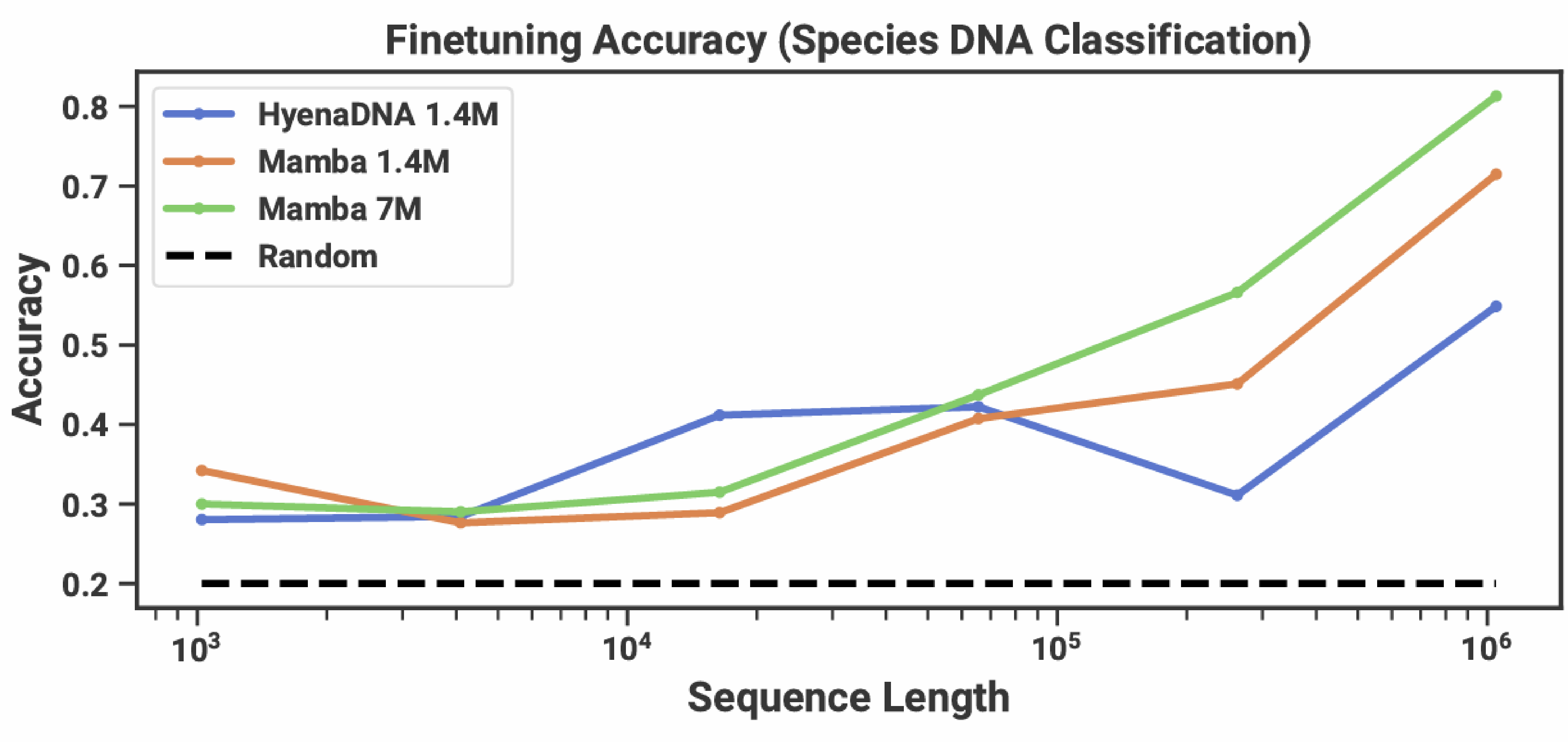
Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Speed



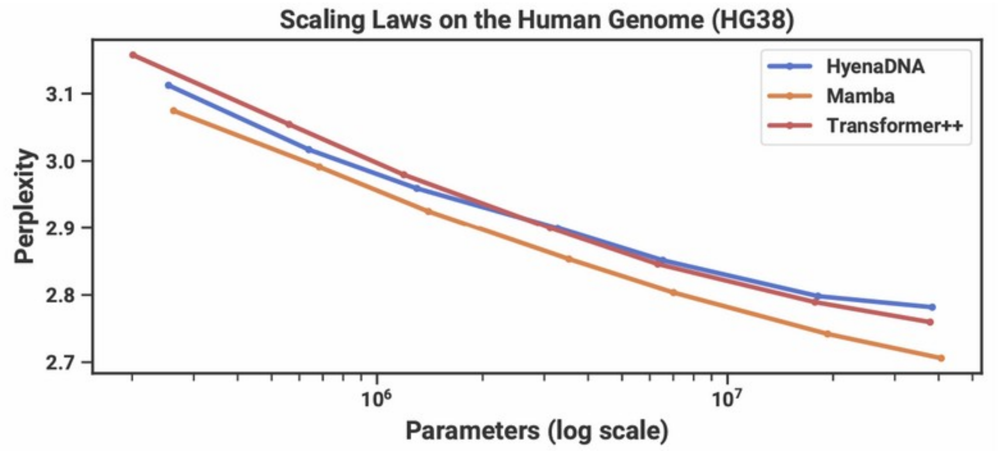
Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

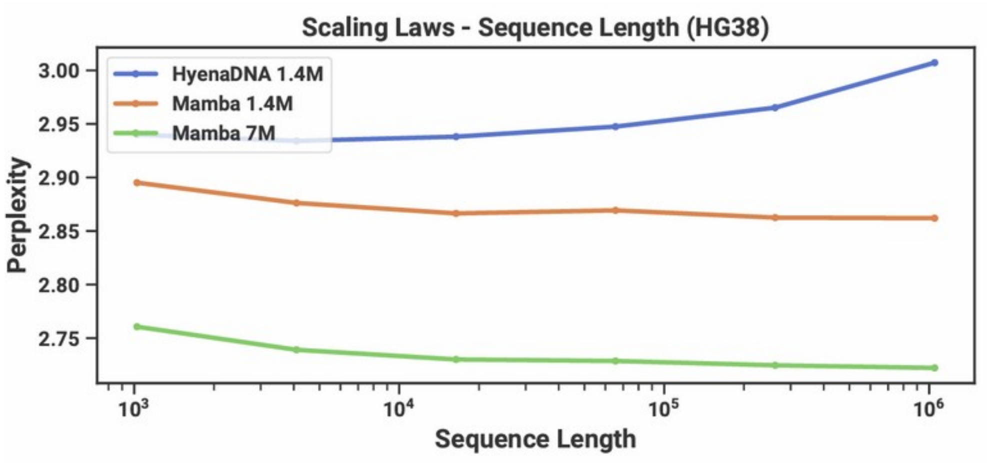
DNA Modeling



Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

DNA Modeling – Scaling laws

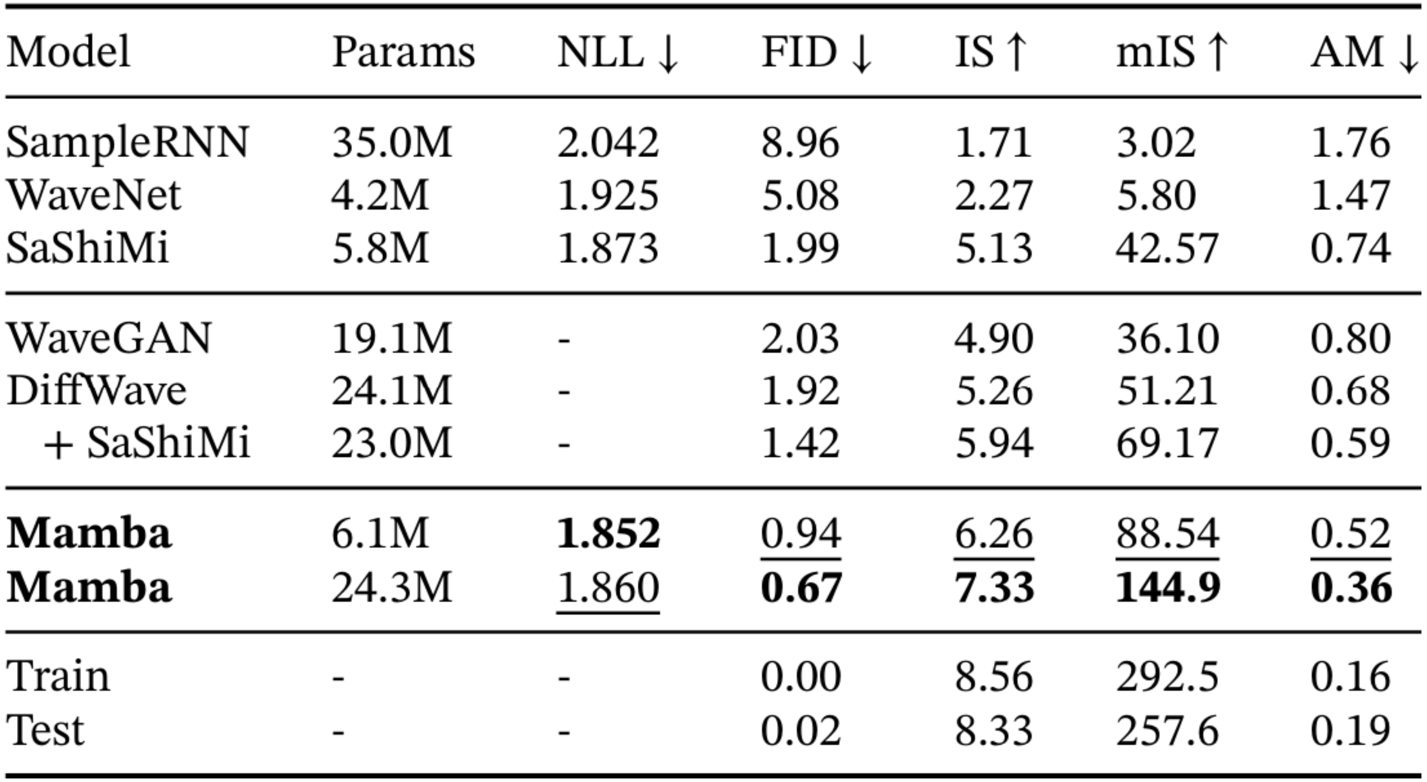




Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Audio Generation

Performance on **SC09**, a speech generation benchmark



Model	Params	NLL ↓	FID ↓	IS ↑	mIS ↑	AM ↓
SampleRNN	35.0M	2.042	8.96	1.71	3.02	1.76
WaveNet	4.2M	1.925	5.08	2.27	5.80	1.47
SaShiMi	5.8M	1.873	1.99	5.13	42.57	0.74
WaveGAN	19.1M	-	2.03	4.90	36.10	0.80
DiffWave	24.1M	-	1.92	5.26	51.21	0.68
+ SaShiMi	23.0M	-	1.42	5.94	69.17	0.59
Mamba	6.1M	**1.852**	0.94	6.26	88.54	0.52
Mamba	24.3M	1.860	**0.67**	**7.33**	**144.9**	**0.36**
Train	-	-	0.00	8.56	292.5	0.16
Test	-	-	0.02	8.33	257.6	0.19

Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Performance Summary

- Excellent performance on synthetic benchmarks
- Matches the performance of transformers in language tasks
- Shows promising scaling laws across all domains

Discussion

Discussion

- Strengths:
  - Demonstrates great speed on long sequences
  - Matches Transformer accuracy
  - Scaling laws look promising

- Weaknesses:
  - empirically evaluated up to 2.4B parameters
  - scaling not yet empirically evaluated for larger sizes

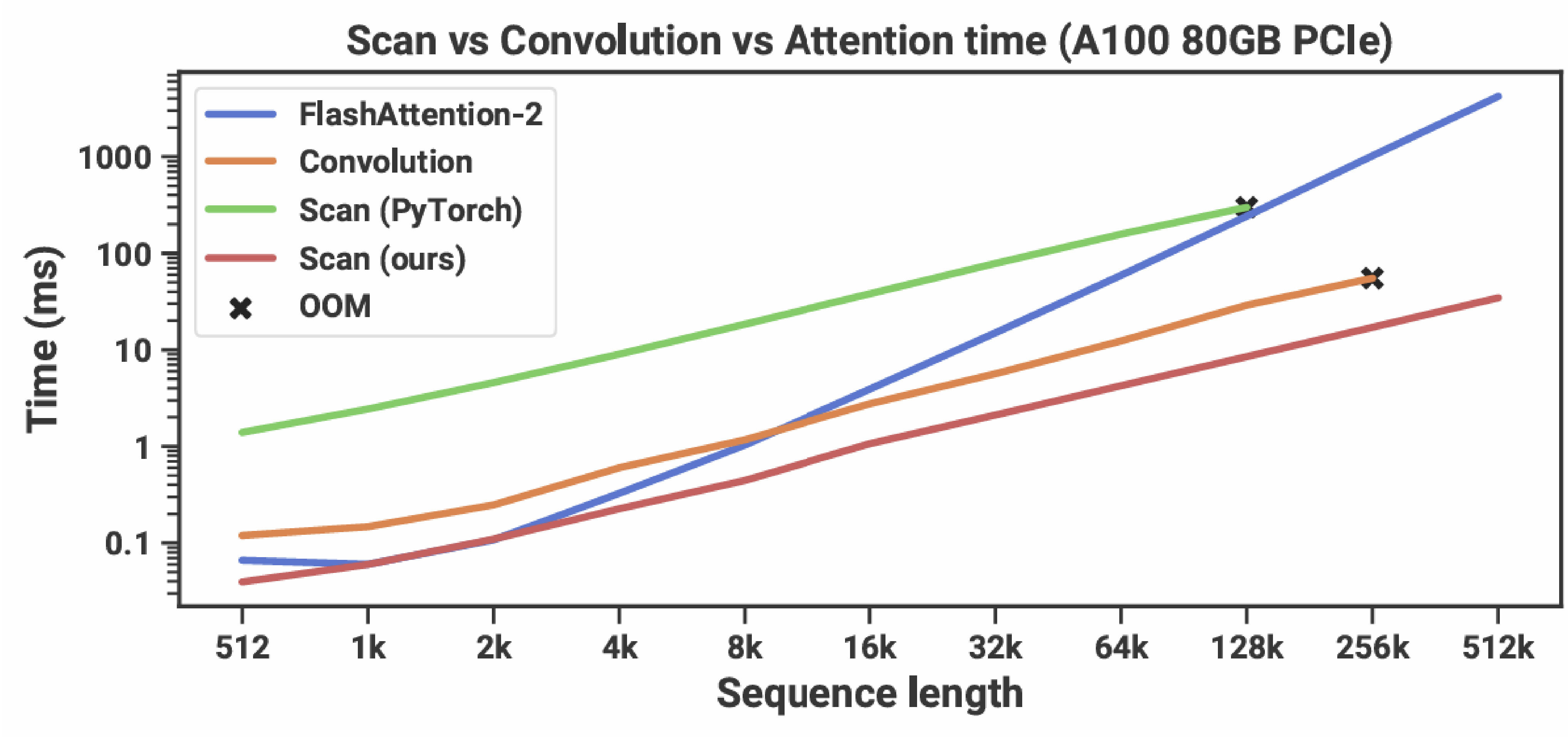
Discussion - ICLR rejection

- “Absence of Results on LRA (Long Range Arena)”
- “Evaluation using perplexity: The reviewer questioned the reliance on perplexity as the major metric for evaluation. ”

Source: https://openreview.net/forum?id=AL1fq05o7H

Thank you!

Speedup due to hardware optimization



Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Discretized state space model - extra

Introduced time step

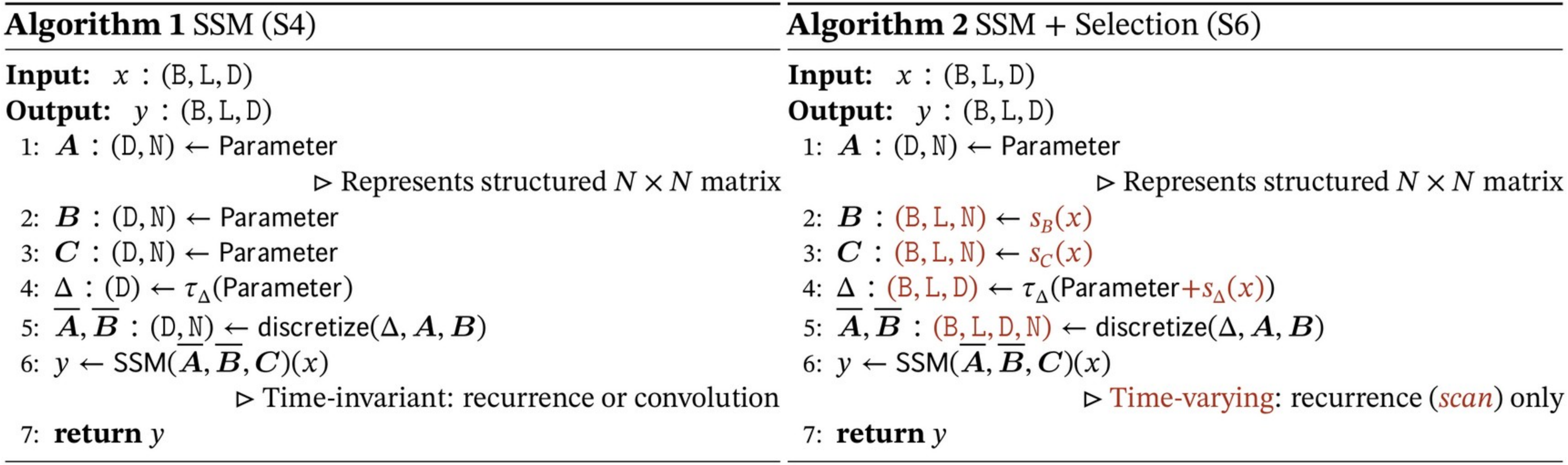
Discretized A and B:

Discretized SSM:

Concrete discretization rule:

Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao

Selective SSM algorithm



Algorithm 1 SSM (S4)

```
Input: $x : (\mathtt{B}, \mathtt{L}, \mathtt{D})$
Output: $y : (\mathtt{B}, \mathtt{L}, \mathtt{D})$
  $\boldsymbol{A} : (\mathtt{D}, \mathtt{N}) \leftarrow$ Parameter
                                  ▷ Represents structured $N \times N$ matrix
  $\boldsymbol{B} : (\mathtt{D}, \mathtt{N}) \leftarrow$ Parameter
  $\boldsymbol{C} : (\mathtt{D}, \mathtt{N}) \leftarrow$ Parameter
  $\Delta : (\mathtt{D}) \leftarrow \tau_\Delta(\text{Parameter})$
  $\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}} : (\mathtt{D}, \mathtt{N}) \leftarrow \text{discretize}(\Delta, \boldsymbol{A}, \boldsymbol{B})$
  $y \leftarrow \text{SSM}(\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}}, \boldsymbol{C})(x)$
                                  ▷ Time-invariant: recurrence or convolution
  return $y$
```

Algorithm 2 SSM + Selection (S6)

```
Input: $x : (\mathtt{B}, \mathtt{L}, \mathtt{D})$
Output: $y : (\mathtt{B}, \mathtt{L}, \mathtt{D})$
  $\boldsymbol{A} : (\mathtt{D}, \mathtt{N}) \leftarrow$ Parameter
                                  ▷ Represents structured $N \times N$ matrix
  $\boldsymbol{B} : (\mathtt{B}, \mathtt{L}, \mathtt{N}) \leftarrow s_B(x)$
  $\boldsymbol{C} : (\mathtt{B}, \mathtt{L}, \mathtt{N}) \leftarrow s_C(x)$
  $\Delta : (\mathtt{B}, \mathtt{L}, \mathtt{D}) \leftarrow \tau_\Delta(\text{Parameter} + s_\Delta(x))$
  $\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}} : (\mathtt{B}, \mathtt{L}, \mathtt{D}, \mathtt{N}) \leftarrow \text{discretize}(\Delta, \boldsymbol{A}, \boldsymbol{B})$
  $y \leftarrow \text{SSM}(\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}}, \boldsymbol{C})(x)$
                                  ▷ Time-varying: recurrence (scan) only
  return $y$
```

ETH zürich

Source: Mamba: Linear-Time Sequence Modeling with Selective State Spaces, by Albert Gu, Tri Dao