Mamba

Linear-Time Sequence Modeling with Selective State Spaces

Introduction

- Structured state space sequence models (SSMs) promising class of architectures for sequence modeling
- Combination of RNNs and CNNs
- Computed very efficiently with linear or near-linear scaling in sequence length
- Many SSMs successful in continuous signal data (audio and vision), but less effective at discrete and information-dense data (text)
- Selective state space models achieve the modeling power of Transformers while scaling linearly in sequence length

- Key limitation of prior models - ability to efficiently select data in an input-dependent manner
- By parameterizing the SSM parameters based on the input, filter out irrelevant information and remember relevant information
- Prior SSMs models must be time- and input-invariant in order to be computationally efficient, but with a hardware-aware algorithm that computes the model recurrently with a scan instead of convolution (3× faster on A100 GPUs)

The Need for Efficient Sequence Modeling

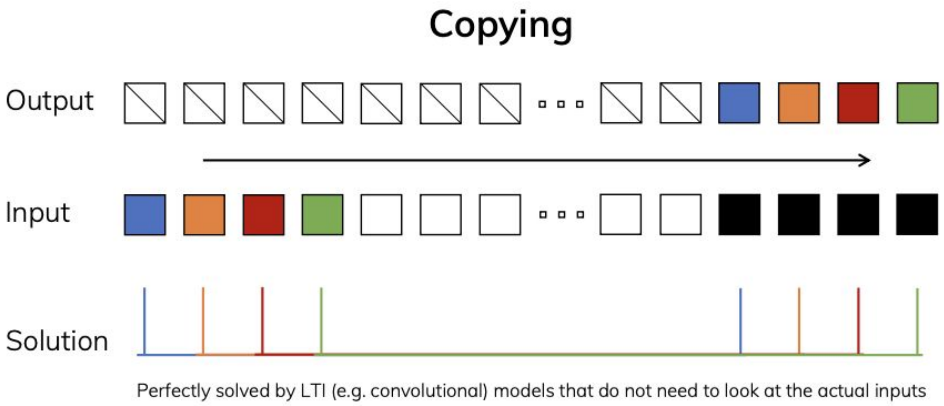
- Traditional Transformer models face computational inefficiencies when handling long sequence data, necessitating the development of more efficient architectures like Mamba.
- Mamba is a novel neural network architecture designed to efficiently handle long sequence data by utilizing selective state space models (SSMs) that scale linearly with sequence length.

Selection as a Means of Compression

- The problem of sequence modeling is compressing context into a smaller state.
- Autoregressive inference requires explicitly storing the entire context, which causes the slow linear-time inference and quadratic-time training of Transformers
- Recurrent models are efficient because they have a finite state, implying constant-time inference and linear-time training.
- Effective models are characterized by how well this state has compressed the context.
- Efficient models are characterized by how well this state has compressed the content.

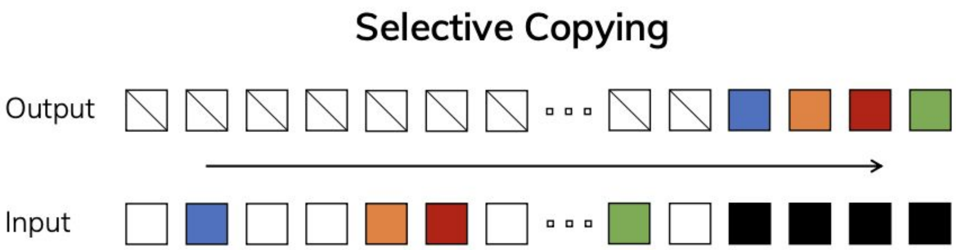
Synthetic Tasks

The standard version of the **Copying task** involves constant spacing between input and output elements and is easily solved by time-invariant models such as linear recurrences and global convolutions.



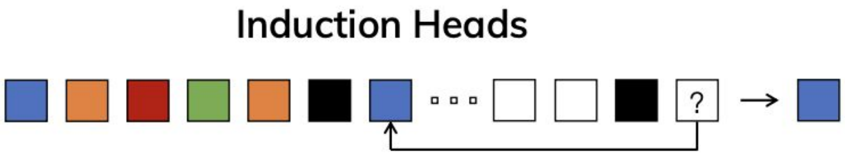
Synthetic Tasks

- The **Selective Copying task** has random spacing in between inputs and requires time-varying models that can selectively remember or ignore inputs depending on their content.
- Requires content-aware reasoning.



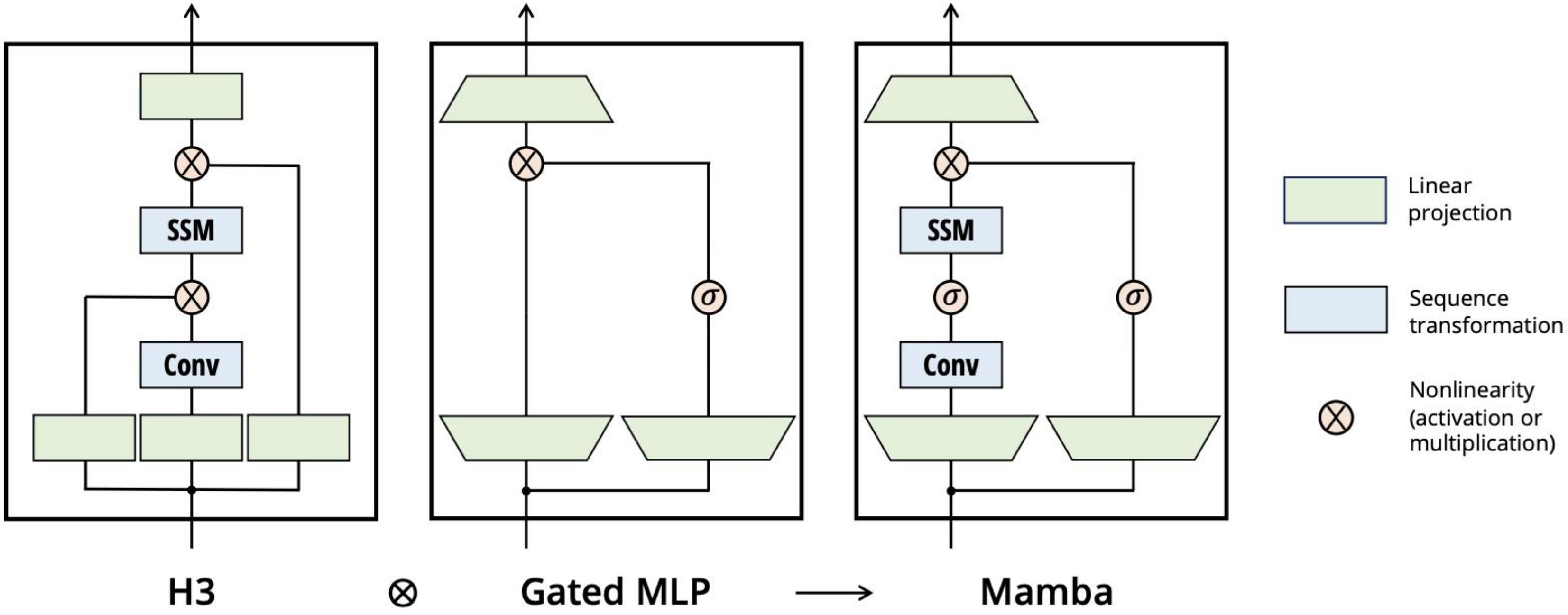
Synthetic Tasks

- The **Induction Heads task** is an example of associative recall that requires retrieving an answer based on context, a key ability for LLMs.
- Requires context-aware reasoning.



Simplified Architecture Design

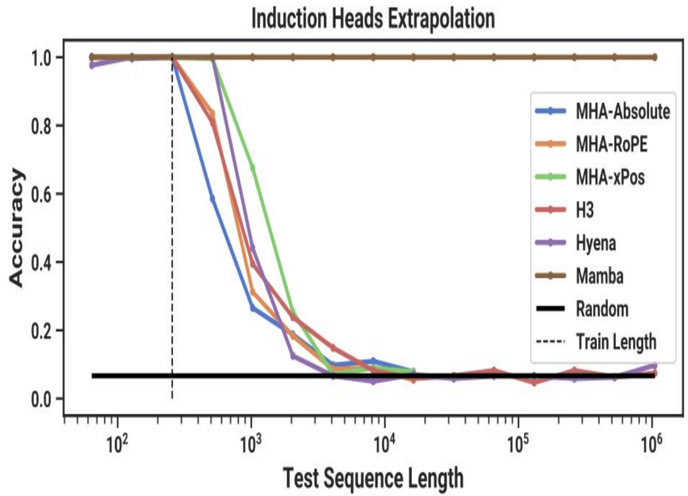
- H3 architecture is the basis for the most SSM architectures
- Mamba combines the H3 block with the ubiquitous MLP block into single block, ie, stacked homogenously, inspired by the gated attention unit (GAU).
- Compared to the H3 block, Mamba replaces the first multiplicative gate with an activation function.
- Compared to the MLP block, Mamba adds an SSM to the main branch
- The number of SSM parameters are much smaller in comparison
- For σ we use the SiLU / Swish activation


SSM
Conv
SSM
Conv
σ
σ
σ
Linear projection
Sequence transformation
Nonlinearity (activation or multiplication)
H3
Gated MLP
Mamba

Empirical Evaluation - Synthetic Tasks

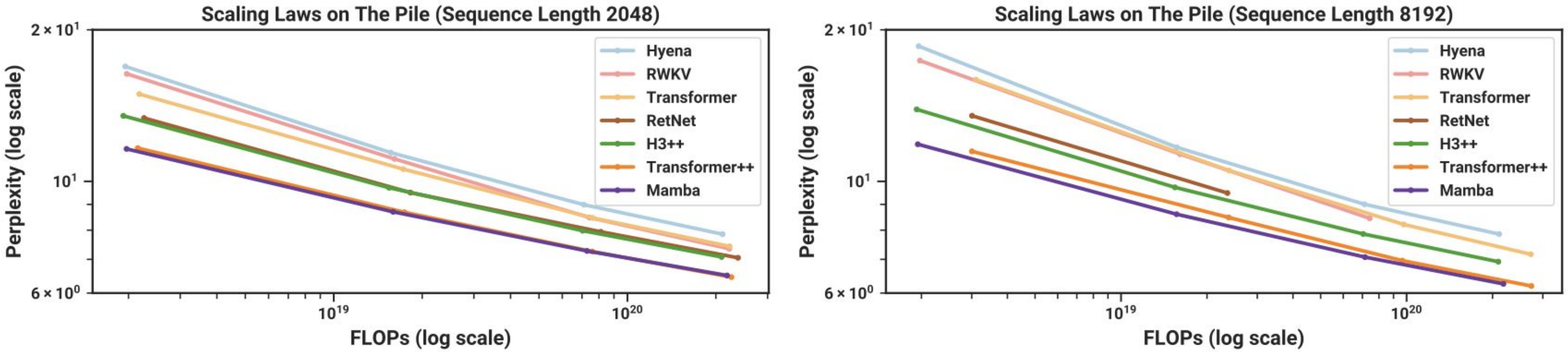
Mamba demonstrates the ability to solve important synthetic tasks such as selective copying and induction heads, extrapolating solutions indefinitely long (>1M tokens).

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**



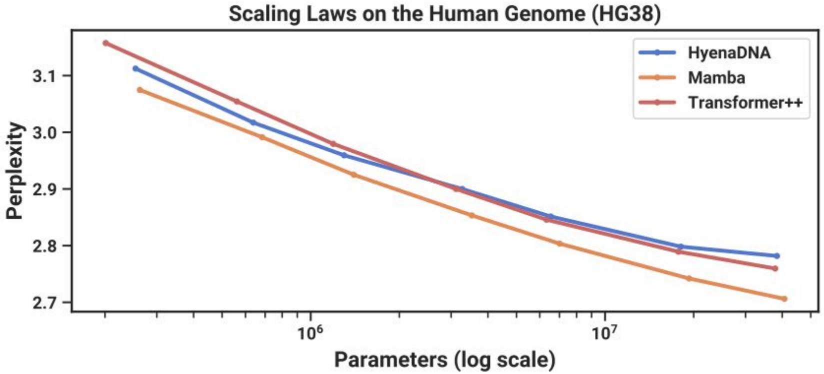
Empirical Evaluation - Language Modeling

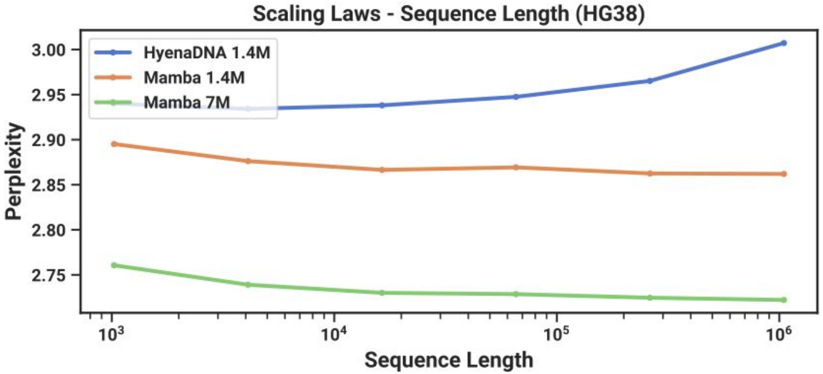
- Mamba achieves Transformer-quality performance in pretraining perplexity and downstream evaluations, with 5× generation throughput compared to Transformers of similar size.
- Mamba scales better than all other models as the sequence length grows.

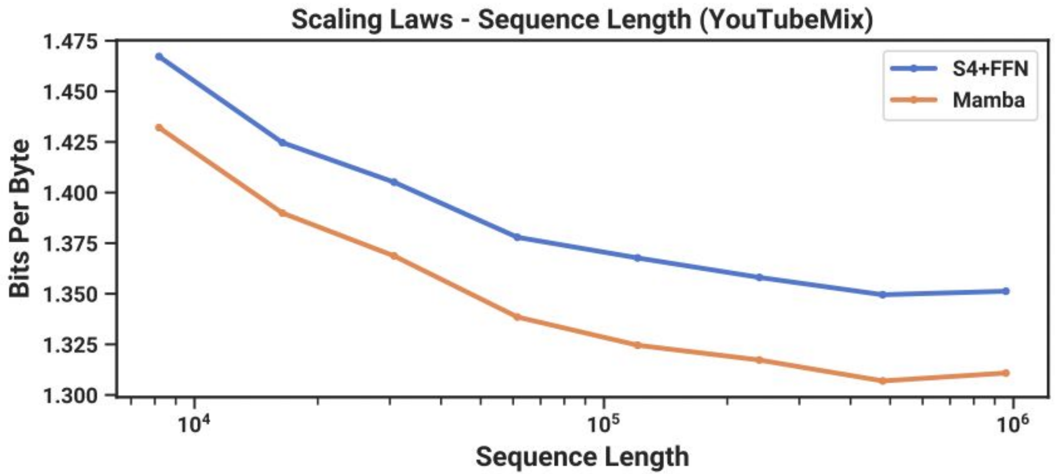


Empirical Evaluation - Audio and Genomics Modeling

- Mamba outperforms prior models on modeling audio waveforms and DNA sequences, demonstrating improved performance with longer context up to million-length sequences.
- Mamba facilitates better performance with increasing context length and model size






Scaling Laws - Sequence Length (YouTubeMix)
S4+FFN
Mamba
Bits Per Byte
1.475
1.450
1.425
1.400
1.375
1.350
1.325
1.300
10^4
10^5
10^6
Sequence Length

Computational Efficiency

- Mamba's efficient scan is 40× faster than a standard implementation during training, and achieves 5× higher throughput than Transformers during inference, making it a practical and efficient choice for large-scale sequence modeling tasks.
- The model's linear scaling in sequence length during training and its ability to handle long contexts effectively position it as a scalable solution for diverse applications.

Future Research

Open-Source Model Code: The authors have open-sourced the model code and pre-trained checkpoints to facilitate further research and application, empowering the broader research community to leverage and build upon the Mamba architecture.

Conclusion

- Mamba perform context-dependent reasoning while scaling linearly in sequence length.
- It matches or exceeds the performance of strong Transformer models.
- Mamba is positioned as a general sequence model backbone with the potential to impact various domains, such as genomics, audio, and video.