Motif-Mamba: network motif improved mamba for long-range sequence modeling

Chonghe Hao[1,2,†] **Yue Sun**[1,†] **Jian Zhang**[1,2,†] **Yansong Wang**[1,2,†]
Wangzi Yao[3] **Yunjie Yao**[1,2] **Tielin Zhang**[1,2,3,4,*]
[1]Center for Excellence in Brain Science and Intelligence Technology,
State Key Laboratory of Brain Cognition and Brain-inspired Intelligence Technology,
Institute of Neuroscience, Chinese Academy of Sciences, Shanghai 200031, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
[3]Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[4]The Key Laboratory of Cognition and Decision Intelligence for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[†]Equal contribution. [*]Corresponding author: zhangtielin@ion.ac.cn

Abstract

Efficient long-sequence modeling remains a central challenge for large language models, as self-attention scales quadratically with sequence length. Mamba offers a linear-time alternative through selective state space recurrence, but its predominantly diagonal state transitions restrict explicit interactions among state dimensions. We propose Motif-Mamba, a structured state space model that augments Mamba with a motif-constrained low-rank recurrent pathway. Inspired by the dynamics of three-node network motifs, the proposed pathway projects hidden states into a compact dynamical subspace, imposes motif-guided interactions, and maps the resulting dynamics back to the original state space. This design enables cross-dimensional communication while preserving linear-time recurrence. Experiments on long-sequence extrapolation, language modeling benchmarks, and brain–computer interface decoding show consistent improvements over Mamba backbones, suggesting that motif-guided low-rank dynamics provide an effective structural prior for long-range sequence modeling.

1 Introduction

Large language models have achieved remarkable progress in natural language understanding, text generation, code reasoning, and multimodal modeling [1]. As their application scenarios expand, effective long-sequence processing becomes increasingly important for tasks such as long-document understanding, multi-turn dialogue, code repository analysis, knowledge retrieval, and scientific time-series modeling [2]. However, most existing large language models are based on the Transformer architecture, whose self-attention mechanism explicitly computes pairwise token interactions and incurs quadratic computational and memory costs with respect to sequence length [3]. Although prior work has explored sparse attention, long-document Transformers, and memory-efficient attention to alleviate this bottleneck [4, 5, 6], efficient long-sequence modeling remains a central challenge.

State space models provide an efficient alternative to attention by propagating information through recurrent state updates [7]. Related recurrent and retention-based architectures have also been explored for efficient sequence modeling [8, 9]. Mamba further introduces input-dependent selection and enables linear-time sequence modeling, making it attractive for long-context modeling [10]. However, its diagonal or nearly diagonal state transitions limit direct interactions among state dimensions, restricting cross-dimensional dynamics and long-range dependency modeling.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

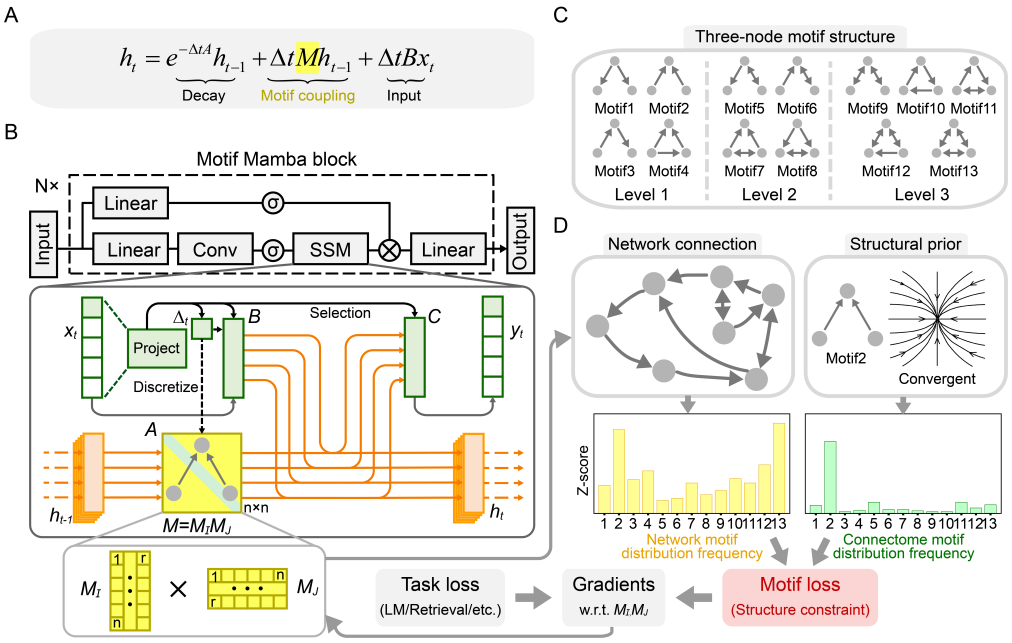


Figure 1: Overview of Motif-Mamba. (A) Motif-Mamba adds a motif-constrained coupling term to the standard Mamba state update. (B) Motif-Mamba architecture. The input is processed through the selective SSM pathway, while the motif-constrained low-rank matrix $M = M_I M_J$ provides an additional recurrent coupling path for efficient cross-dimensional state interaction. (C) Three-node motifs are used to describe local connectivity patterns. (D) Motif statistics from the learned matrix are compared with target motif statistics to form the motif loss, which jointly optimizes the structural constraint and the task objective.

Network motifs are recurring local connectivity patterns in complex networks and have been widely studied as functional building blocks of structured systems [11]. Recent work shows that 13 directed three-node motifs can be grouped into a three-level dynamical hierarchy, where different motif groups correspond to different stability and flexibility regimes [12, 13, 14]. This suggests that motifs are not only static graph patterns, but also carry interpretable dynamical properties. Moreover, motif-based analysis provides a compact way to link local connectivity patterns with higher-level network function [15]. Motivated by this view, we introduce motif-based structural priors into Mamba, aiming to guide hidden state interactions with more organized and interpretable dynamical patterns.

To efficiently incorporate this motif prior, we propose Motif-Mamba, a structured state space model that implements motif-constrained state interaction through a low-rank dynamical pathway. Specifically, Motif-Mamba projects the high-dimensional hidden state into a compact low-dimensional space, performs motif-constrained interaction in that space, and maps the result back to the original state space. This design enables structured cross-dimensional interaction without using a dense transition matrix, while remaining consistent with prior evidence that low-rank structures can reveal compact neural dynamics [16, 17].

By introducing motif-constrained low-rank dynamics, Motif-Mamba encourages more organized state interactions and turns hidden state evolution into a more structured gray-box dynamical process. We evaluate Motif-Mamba on long-sequence extrapolation, standard language model benchmarks, ablation studies, and brain–computer interface decoding tasks. The results show that Motif-Mamba improves long-range dependency modeling while preserving the computational efficiency of Mamba.

Our primary contributions can be summarized as follows:

- We propose Motif-Mamba, a motif-constrained state space model that enhances Mamba for efficient long-sequence modeling.
- We design a low-rank recurrent pathway that embeds motif-guided state interactions into Mamba while preserving linear-time sequence modeling.

- We evaluate Motif-Mamba on long-sequence extrapolation, standard language model benchmarks and brain–computer interface decoding, demonstrating consistent improvements over Mamba and other baselines.
- We analyze different motif constraints from the dynamics of motif and study how motifs with different dynamical properties affect hidden-state evolution.

2 Related Work

2.1 State Space Models for Sequence Modeling

State space models (SSMs) enable efficient sequence modeling. Unlike attention-based models that explicitly compute pairwise token interactions, SSMs model dependencies through recurrent state evolution and can be implemented efficiently in recurrent or convolutional forms. Representative models such as S4, S5, and H3 have shown strong performance in long-range sequence modeling and language modeling [7, 18, 19].

Mamba further advances this direction by introducing selective state space models with input-dependent parameters and a hardware-aware parallel scan algorithm [10]. This design enables linear scaling with sequence length and makes Mamba suitable for efficient long-context modeling. However, Mamba still relies heavily on diagonal or element-wise state transitions, which limits explicit interaction among hidden state dimensions. This motivates efficient cross-dimensional interaction through structured low-rank dynamics.

2.2 Low-Rank Modeling in Deep Learning

Low-rank decomposition is widely used for model compression, acceleration, and efficient adaptation. Factorizing matrices into lower-rank products reduces parameter and computational costs while retaining expressive capacity [16]. In Transformer-based models, Linformer reduces the complexity of self-attention through low-rank approximation, while LoRA injects trainable low-rank matrices for parameter-efficient adaptation [20, 21].

Low-rank structures are also closely related to compact dynamical representations. In recurrent neural networks, low-rank connectivity has been shown to induce low-dimensional dynamics that link network structure, state evolution, and computation [17]. However, most existing low-rank methods are generic and do not explicitly impose structural constraints on the learned interaction. Motif-Mamba addresses this limitation by introducing motif-constrained low-rank coupling into the Mamba state update.

2.3 Network Motifs and Structured Priors

Network motifs are recurrent and statistically over-represented connectivity patterns among small groups of nodes in complex networks [11]. Such local structures have been studied in biological, ecological, and engineered systems. In neuroscience, connectome studies have revealed structured connectivity patterns in the mouse brain, suggesting that local circuit organization may support efficient information processing [22, 23].

Motif information has also been used in graph learning to capture higher-order structural patterns beyond pairwise edges. Motif-based graph neural networks have shown that incorporating motif-level information can improve representation learning on graph-structured data [24, 25]. However, motif-based structural priors remain underexplored in efficient sequence models such as Mamba. Motif-Mamba integrates motif-constrained low-rank dynamics into selective state space models to enhance structured cross-dimensional interactions while preserving linear-time sequence modeling.

3 Method

3.1 Preliminaries: Selective State Space Models

State space models describe sequence dynamics through recurrent hidden state updates and have been widely used for efficient long-range sequence modeling [7, 10]. Given an input sequence $x(t)$,

a continuous-time state space model can be written as

$$\frac{dh(t)}{dt} = -A(t)h(t) + B(t)x(t), \tag{1}$$

where $h(t)$ denotes the hidden state and $A(t) \in \mathbb{R}^{n\times n}$ is the general state transition matrix. After discretization, the state update can be expressed as

$$h_t = \bar{A}_t h_{t-1} + \bar{B}_t x_t, \tag{2}$$

where $\bar{A}_t$ and $\bar{B}_t$ denote the discretized transition and input terms respectively.

Mamba extends classical state space models with a selective mechanism where several parameters are dynamically generated from the input. In particular, the input-dependent step size Δ_t, input projection B_t, and output projection C_t allow the model to selectively preserve or forget information according to the current token. A simplified Mamba state update can be written as

$$h_t = \exp(-\Delta_t A_t)h_{t-1} + \Delta_t B_t x_t. \tag{3}$$

This recurrent formulation enables linear-time sequence processing. However, to maintain computational efficiency, A is usually parameterized as a diagonal matrix in practical SSM architectures. As a result, different state dimensions lack explicit coupling and tend to evolve independently. Although efficient, this structure limits the ability of the model to represent complex high-dimensional dynamics.

3.2 Low-Rank Recurrent Coupling

Motif-Mamba addresses this limitation by introducing a low-rank recurrent interaction pathway into the selective SSM. Let $M_I \in \mathbb{R}^{n\times r}$ and $M_J \in \mathbb{R}^{r\times n}$ denote the adapter matrices where n is the hidden-state dimension and $r \ll n$ is the low-rank dimension. We consider the continuous-time SSM with this low-rank structure defined by the following differential equation

$$\frac{dh(t)}{dt} = -Ah(t) + M_I M_J h(t) + B(t)x(t). \tag{4}$$

In this formulation, the term $M_I M_J h(t)$ introduces recurrent interaction among state dimensions through a compact low-dimensional space. The state is projected to $z(t) = M_J h(t)$ and mapped back via $M_I z(t)$, which drives the recurrent interaction.

Lemma 1. First-order semi-exponential discretization with low-rank interaction. Freezing the input-dependent parameters within each token interval and approximating the exponential-Euler driving coefficient by $\Delta_t I$ yields the following first-order discrete-time approximation to Eq. 4

$$h_t = e^{-\Delta_t A}h_{t-1} + \Delta_t M_I M_J h_{t-1} + \Delta_t B_t x_t. \tag{5}$$

The local truncation error is $O(\Delta_t^2)$ under bounded matrix norms, states, and inputs. The detailed derivation is provided in Appendix B.1. This formulation shows that the low-rank term acts as a recurrent driving component inside the SSM dynamics.

The update in Eq. 5 possesses several key properties. First, $M_I M_J$ is a square recurrent coupling matrix with rank at most r allowing it to encode structured state interactions efficiently. Second, the product $M_I M_J h_{t-1}$ is evaluated as $M_I(M_J h_{t-1})$ which introduces cross-state coupling within a compact latent manifold. Third, while the diagonal self-decay term $e^{-\Delta_t A}h_{t-1}$ is preserved, the low-rank term $\Delta_t M_I M_J h_{t-1}$ injects structured cross-state information flow. Motif-Mamba thus introduces explicit recurrent interaction into Mamba while maintaining the efficient diagonal SSM backbone.

3.2.1 Complexity Analysis

The low-rank pathway enables cross-dimensional interaction without a dense transition matrix. A dense recurrent coupling Wh_{t-1} with $W \in \mathbb{R}^{n\times n}$ requires $O(n^2)$ computation per token. In contrast, Motif-Mamba computes $M_I(M_J h_{t-1})$ with a complexity of $O(nr)$. Since $r \ll n$, the low-rank

coupling is significantly cheaper. For a sequence of length L, the recurrent complexity comparison is as follows

$$\begin{cases} \text{Standard Mamba:} & O(Ln), \\ \text{Motif-Mamba:} & O(Ln) + O(Lnr), \\ \text{Dense recurrent coupling:} & O(Ln^2). \end{cases} \tag{6}$$

Thus, Motif-Mamba adds explicit cross-state interaction while preserving the linear-time advantage of selective SSMs when r is small. Empirical throughput and latency are reported in Appendix D.5.

3.2.2 Recovering Hidden Dynamics from Latent States

The latent state $z_t = M_J h_t$ defines the low-rank recurrent component $M_I z_t$ that enters the state update. Because this compact variable drives the cross-dimensional interaction, analyzing the system within this latent space provides two distinct analytical advantages.

First, z_t establishes a rigorous mathematical basis for recovering the high-dimensional hidden-state trajectory. The following reconstruction property explains how the temporal evolution of z_t captures the structural information of the complete system.

Lemma 2. Reconstruction from latent states and inputs. Consider the discrete Motif-Mamba recurrence where $z_{t-1} = M_J h_{t-1}$. Given the initial state h_0, the history of latent states $\{z_{\tau-1}\}_{\tau=1}^{t}$, the input history $\{x_\tau\}_{\tau=1}^{t}$ and the transition parameters $\{A_\tau, \Delta_\tau, B_\tau\}_{\tau=1}^{t}$, the hidden state h_t is determined by

$$h_t = \Gamma(t,0)h_0 + \sum_{\tau=1}^{t} \Gamma(t,\tau)\left(\Delta_\tau M_I z_{\tau-1} + \Delta_\tau B_\tau x_\tau\right), \tag{7}$$

where $\Gamma(t,\tau) = \prod_{j=\tau+1}^{t} e^{-\Delta_j A_j}$ and $\Gamma(t,t) = I$. The proof is provided in Appendix B.2.

This lemma mathematically guarantees that the sequence of z_t along with the input history captures the complete recurrent contribution mediated by the low-rank pathway.

Second, the low-dimensional z_t provides a coordinate system for visualizing the projected dynamics. With external input removed and transition parameters frozen, define the effective discrete transition $T = e^{-\Delta A} + \Delta M_I M_J$. Using the local pseudoinverse lifting $h \approx M_J^+ z$, the projected one-step displacement is

$$v(z) = M_J(T - I)M_J^+ z. \tag{8}$$

Here M_J^+ is the pseudoinverse of M_J. For full row rank M_J, this equals $M_J T M_J^+ z - z$. This field is an approximate projected discrete vector field: components of h in the null space of M_J are not recovered by the lifting. It complements the history-dependent reconstruction in Lemma 2 rather than defining an exact closed latent dynamical system.

3.3 Motif-Constrained Structural Prior

Network motifs act as functional building blocks that govern the stability and flexibility of recurrent interactions. To guide the hidden state evolution toward these interpretable dynamical regimes, we treat the low-rank coupling matrix $M = M_I M_J$ as a structurally constrained optimization object.

Calculating motif frequencies relies on a discrete adjacency structure which presents a challenge for backpropagation during gradient optimization. Following previous motif-guided optimization strategies [12, 13], we overcome this by introducing a continuous relaxation and substituting the binary adjacency matrix with a differentiable proxy $\tilde{M} \approx \sigma(\beta(M \odot M - \theta))$ where $\odot$ denotes the Hadamard product, $\sigma(\cdot)$ is the sigmoid function and β is a temperature parameter controlling the sharpness of the threshold. Using this differentiable proxy, we extract the learned motif frequency vector

$$\mathbf{m}_M = \mathcal{M}(\tilde{M}), \tag{9}$$

where $\mathcal{M}(\cdot)$ represents the matrix-based motif extraction operator detailed in Appendix C. Given a target connectivity matrix representing the desired dynamical regime, we similarly compute its target motif frequency vector $\mathbf{m}_{\text{target}}$.

We measure the structural alignment between the learned and target motif profiles using the coefficient of determination

$$R^2 = 1 - \frac{\sum_{k=1}^{K} \left(m_{M,k} - m_{\text{target},k}\right)^2}{\sum_{k=1}^{K} \left(m_{\text{target},k} - \bar{m}_{\text{target}}\right)^2 + \epsilon}, \tag{10}$$

where K denotes the number of selected motif categories and ϵ provides numerical stability. The motif loss is then defined as the deviation from perfect alignment

$$\mathcal{L}_{\text{motif}} = 1 - R^2. \tag{11}$$

This differentiable formulation enables targeted motif distributions to be reliably embedded into the network structure.

During training, the final objective function combines the standard task loss $\mathcal{L}_{\text{task}}$ with the motif-based structural regularization

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda \mathcal{L}_{\text{motif}}, \tag{12}$$

where λ controls the strength of the structural constraint. This joint optimization ensures that the learned representation satisfies the specific sequence modeling requirements while maintaining a motif-guided dynamical organization.

4 Experiments

4.1 Long-Sequence Extrapolation

To evaluate long-sequence extrapolation, we use the induction heads task, a synthetic associative recall task that requires the model to remember a previous cue–target association and copy the corresponding continuation when the cue reappears. This tests long-context memory in a controlled setting.

Unless otherwise specified, Motif-Mamba denotes the Motif-2-constrained variant. This selection is derived from the three level motif hierarchy where different three node configurations correspond to distinct structural and dynamical regimes [12, 14]. The topological features of Motif 2 are characterized by two nodes converging toward a single target without any reciprocal edges or closed loops. In contrast Motif 12 incorporates reciprocal connections and forms a cyclic loop structure. Previous dynamical analyses indicate that these topological differences fundamentally alter the stability of the recurrent interaction [12, 13, 14]. The reciprocal edges and cycles in Motif 12 create feedback loops that are harder to stabilize than the convergent structure of Motif 2. This structural complexity makes Motif 12 inherently more difficult to optimize during training.

All models are trained with sequence length 256 and vocabulary size 16, and are evaluated on sequence lengths from 64 to 16384 tokens. We compare Motif-Mamba with representative sequence modeling baselines, including LSTM [26], Transformer [3], RWKV [8], and standard Mamba [10]. We use the same data generation process, optimizer, batch size, learning rate schedule, and training steps across models. The metric is next-token prediction accuracy at induction target positions.

Figure 2 shows that Motif-Mamba maintains stronger extrapolation performance as the test length increases. Compared with standard Mamba and other baselines, Motif-Mamba is more stable beyond the training context, suggesting that motif-constrained low-rank dynamics help preserve and retrieve task-relevant information over long sequences. The numerical results corresponding to the curves in Figure 2 are provided in Table S7 as a complementary quantitative summary.

4.2 Language Model Evaluation

We further evaluate the general language modeling ability of Motif-Mamba on standard benchmarks from the lm-evaluation-harness [27], including LAMBADA [28], HellaSwag [29], PIQA [30], ARC-E and ARC-C [31], and WinoGrande [32]. These tasks evaluate language modeling, commonsense and physical reasoning, and multiple-choice question answering beyond the synthetic setting. Figure 3 shows a qualitative long-context QA example, where Motif-Mamba retrieves the relevant evidence and produces the correct answer, while standard Mamba fails to preserve the task-relevant information.

We compare Motif-Mamba with standard Mamba and other representative baseline models of similar scale. The evaluation metrics include LAMBADA perplexity and accuracy-based scores for the

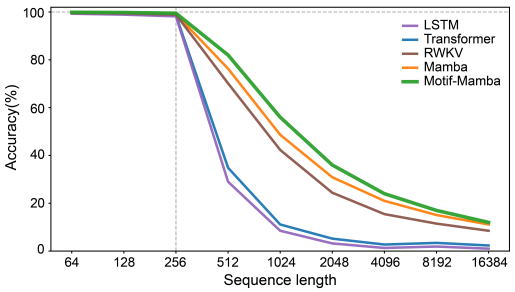


Figure 2: Long-sequence extrapolation on the induction heads task. Models are trained with 256 tokens and tested from 64 to 16384 tokens.

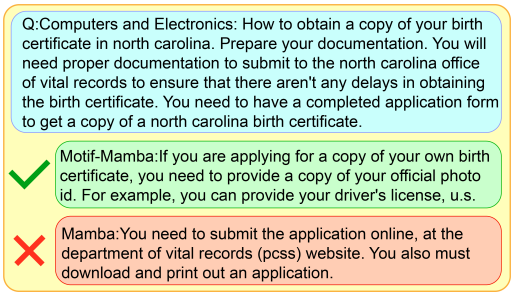


Figure 3: Illustrative example of Motif-Mamba performing a long-context question answering task.

Table 1: Language model evaluation results on standard benchmarks (%). Mamba accuracy scores are obtained after continued pretraining with the same additional 10M tokens, data order, tokenizer, sequence length, and optimization settings as Motif-Mamba.

Model	LAMB. PPL ↓	LAMB. ACC ↑	HellaS. ACC ↑	PIQA ACC ↑	Arc-E ACC ↑	Arc-C ACC ↑	WinoGr. ACC ↑	Average ACC ↑
Hybrid H3-130M [19]	89.48	25.8	31.7	64.2	44.4	24.2	50.5	40.1
Pythia-160M [33]	38.10	33.0	30.2	61.4	43.2	24.1	51.9	40.6
Mamba-130M [10]	16.07	44.3	35.3	64.5	48.0	24.3	51.9	44.7
Motif-Mamba-130M	**15.47**	**44.7**	**35.7**	**64.7**	**48.5**	**24.6**	**52.4**	**45.1**
Hybrid H3-360M [19]	12.58	48.0	41.5	68.1	51.4	24.7	54.1	48.0
Pythia-410M [33]	10.84	51.4	40.5	66.9	52.1	24.6	53.8	48.2
Mamba-370M [10]	8.14	55.6	46.4	69.5	55.1	28.0	55.3	51.6
Motif-Mamba-370M	**7.95**	**55.9**	**47.1**	**70.0**	**55.6**	**28.7**	**56.3**	**52.3**
Pythia-1B [33]	7.92	56.1	47.2	70.7	57.0	27.1	53.5	51.9
Mamba-790M [10]	6.02	62.7	55.1	72.1	61.2	29.6	56.1	56.1
Motif-Mamba-790M	**5.88**	**62.8**	**55.9**	**72.6**	**61.5**	**29.6**	**57.1**	**56.6**
GPT-Neo 1.3B [34]	7.50	57.2	48.9	71.1	56.2	25.9	54.9	52.4
Hybrid H3-1.3B [19]	11.25	49.6	52.6	71.3	59.2	28.1	56.9	53.0
OPT-1.3B [35]	6.64	58.0	53.7	72.4	56.7	29.6	59.5	55.0
Pythia-1.4B [33]	6.08	61.7	52.1	71.0	60.5	28.5	57.2	55.2
RWKV-1.5B [8]	7.04	56.4	52.5	72.4	60.5	29.4	54.5	54.3
Mamba-1.4B [10]	5.04	64.9	59.0	74.2	65.5	32.9	61.5	59.7
Motif-Mamba-1.4B	**4.82**	**65.5**	**60.5**	**75.1**	**66.8**	**33.9**	**62.7**	**60.8**

remaining tasks. Unless otherwise specified, all models are evaluated under the same protocol. In the main experiments, Motif-Mamba uses the default Motif-2 constraint.

Table 1 reports the language model evaluation results. Motif-Mamba consistently improves over the corresponding Mamba backbone across different model sizes. The gains are observed in both LAMBADA perplexity and average accuracy, suggesting that motif-constrained low-rank state interaction not only benefits synthetic long-sequence extrapolation, but also improves general language modeling and reasoning performance.

4.3 BCI Decoding

We evaluate Motif-Mamba on brain–computer interface decoding tasks using the JangoBCI dataset. This dataset contains neural and behavioral recordings from a non-human primate performing an isometric wrist task, and has been used for evaluating long-term BCI decoding and neural alignment methods [36]. Neural signals are naturally long temporal sequences and often contain dependencies across both time and recording channels. Therefore, BCI decoding provides a useful testbed for evaluating whether the proposed structured low-rank dynamics can generalize to non-text sequential data.

The model predicts behavioral or task-related outputs from neural signals. We compare Motif-Mamba with standard Mamba and commonly used sequence models, including LSTM and Transformer. All models are trained and evaluated using the same data split, preprocessing pipeline, and evaluation protocol.

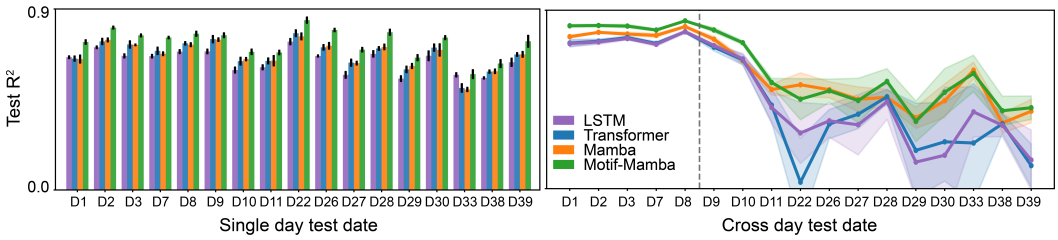


Figure 4: BCI decoding results on the JangoBCI dataset. The figure compares Motif-Mamba with baseline sequence models on long temporal neural signals collected across recording sessions from Day 1 (D1) to Day 39 (D39). The first panel reports single-day training and single-day testing performance, where error bars denote std. The second panel reports cross-day generalization, where data before the dashed line are used for training and data after the dashed line are used only for evaluation; shaded regions denote std ($n = 5$).

Figure 4 shows the decoding performance of different models. Motif-Mamba achieves better decoding performance than the baseline models, suggesting that motif-constrained low-rank dynamics can help capture long-range temporal dependencies in neural time series. These results support extending Motif-Mamba to temporal signal decoding.

4.4 The Role of Motif-Constrained Low-Rank Dynamics

We first conduct ablation studies to examine the role of motif constraints in Motif-Mamba. We separate the effects of low-rank coupling and structural constraints. We compare continued-pretrained Mamba, unconstrained Motif-Blank-Mamba, Motif-2-, Motif-7-, and Motif-12-Mamba, and three additional controls. Motif-Random uses an Erdős–Rényi graph with the same sparsity as Motif-2; Motif-Shuffled mixes the Motif-2 and Motif-12 targets; Norm-Matched removes the motif constraint while matching the layerwise interaction spectral norm of Motif-2. All variants use the 130M backbone and the same additional 10M-token budget. Results for all thirteen motifs are provided in Appendix D.4.

Table 2: Ablation results on standard benchmarks (%).

Model	LAMB. PPL ↓	LAMB. ACC ↑	HellaS. ACC ↑	PIQA ACC ↑	Arc-E ACC ↑	Arc-C ACC ↑	WinoGr. ACC ↑	Average ACC ↑
Mamba-130M	16.07	44.3	35.3	64.5	48.0	24.3	51.9	44.7
Motif-Blank-Mamba-130M	15.79	44.3	35.2	64.5	48.0	24.5	52.3	44.8
Motif-2-Mamba-130M	**15.47**	**44.7**	**35.7**	**64.7**	**48.5**	**24.6**	**52.4**	**45.1**
Motif-7-Mamba-130M	15.71	43.8	35.1	64.5	48.2	24.5	51.9	44.7
Motif-12-Mamba-130M	15.88	44.2	35.5	64.6	48.4	24.5	51.9	44.9
Motif-Random-Mamba-130M	15.90	44.4	35.5	64.4	47.9	24.1	52.0	44.7
Motif-Shuffled-Mamba-130M	15.97	44.4	35.3	64.3	48.2	24.1	**52.4**	44.8
Norm-Matched-Mamba-130M	16.02	44.3	35.3	64.4	47.9	**24.6**	**52.4**	44.8

Table 2 reports the ablation results on standard language model benchmarks. Compared with Mamba-130M, Motif-Blank-Mamba-130M shows almost no improvement as the average accuracy only increases from 44.7% to 44.8%. This suggests that simply adding an unconstrained low-rank pathway is insufficient. In contrast, motif-constrained variants achieve clearer gains which indicates the importance of structural constraints. Among them, Motif-2-Mamba-130M performs best and achieves the highest average accuracy of 45.1%.

We analyze the low-rank recurrent dynamics to examine the effect of structural constraints. Lemmas 1 and 2 establish z_t as the recurrent driving state and a coordinate for history-dependent reconstruction, motivating its use in population dynamics analysis.

Following the theoretical framework from the Method section, we constructed the projected autonomous flow fields $v(z)$ in the rank-2 z-plane (Fig. 5). Although the full high-dimensional hidden state h may contain components in the null space of M_J, the continuations without input initialized from real trial states closely followed the local flow directions. This suggests that the low-dimensional field effectively captures the autonomous component of the low-rank recurrent interaction. Our analy-

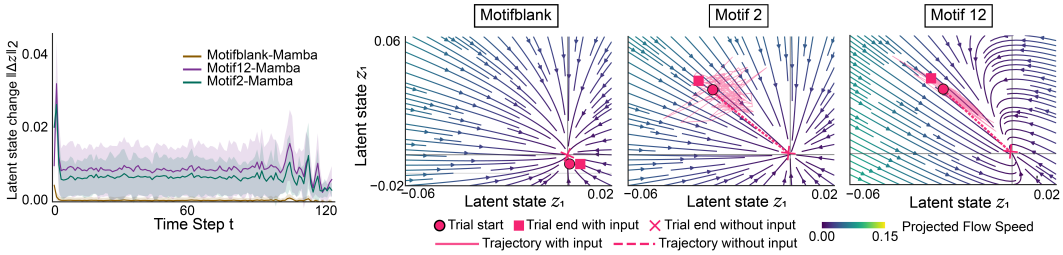


Figure 5: Motif constraints induce structured low-rank recurrent dynamics. Projected flow fields and latent trajectories in the rank-2 z-plane for Motif-Blank, Motif-2 and Motif-12. Solid pink lines denote empirical trajectories with external inputs while dashed pink lines denote autonomous continuations without input. The background vector field indicates the projected autonomous flow $v(z)$ colored by its local speed.

sis reveals that motif-constrained structures induce more pronounced low-rank recurrent interactions compared to the unconstrained Motif-Blank architecture. In Motif-Blank, projected flow and empirical trajectories cluster near the origin, indicating weak latent recurrent interactions. Motif-2 and Motif-12 instead exhibit distinct z-plane trajectories and larger flow magnitudes.

To quantify this structural displacement, we measured the M_I-normalized latent state change $\|\Delta z_{norm}(t)\|_2 = \|z_{norm}(t) - z_{norm}(t-1)\|_2$, where $z_{norm,a}(t) = \|M_{I,:,a}\|_2 (M_J)_a h_{t-1}$. Motif-2 and Motif-12 demonstrated substantially higher $\|\Delta z_{norm}(t)\|_2$ than Motif-Blank across 200 LAMBADA examples which confirms that motif constraints activate stronger and more structured low-rank recurrent interactions. Finally, we compared the interaction magnitudes of Motif-2 and Motif-12 across all layer 0 channels (Fig. 6). We calculated the average $\|\Delta z_{norm}\|_2$ over 200 LAMBADA examples for each channel. Most points fall below the identity line, indicating that Motif-12 generally induces larger latent state changes than Motif-2 across channels. These findings align with previous motif hierarchy studies which suggested that different three-node motif classes are associated with distinct dynamical regimes [12, 14]. Consequently, motif constraints do not merely activate the low-rank pathway, but precisely modulate the intensity and structure of the induced latent dynamics across channels.

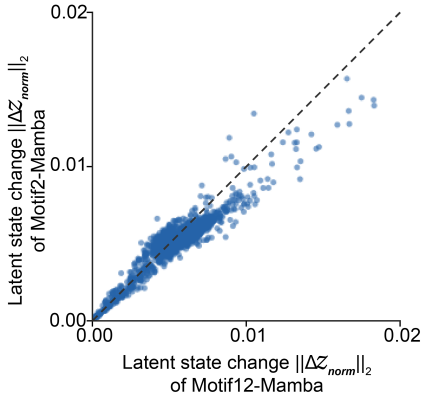


Figure 6: Population-level comparison of latent interaction magnitudes. Each point represents the average $\|\Delta z_{norm}\|_2$ over 200 trials for an individual layer internal channel ($n = 1531$). The coordinates compare the interaction strength of Motif-12-Mamba (x-axis) against Motif-2-Mamba (y-axis). The dashed diagonal indicates the identity line where values are equal. The heavy concentration of points below this line demonstrates that Motif-12 consistently induces stronger low-rank recurrent interactions than Motif-2.

5 Conclusion and Discussion

Conclusion. We introduced Motif-Mamba, a structured extension of Mamba designed to strengthen long-sequence modeling through motif-constrained state dynamics. The proposed motif-guided recurrent pathway enables efficient cross-dimensional state interaction while preserving the linear-time modeling advantage of Mamba. Experiments on long-sequence extrapolation, language model benchmarks, and BCI decoding show that Motif-Mamba improves both text and non-text temporal sequence modeling. Ablation results further demonstrate that the gains mainly come from the motif-based structural constraint rather than from unconstrained low-rank coupling alone.

Discussion. Language modeling gains across multiple Mamba scales suggest that motif-constrained dynamics extend beyond synthetic tasks. Language modeling requires coordinated integration of

lexical, syntactic, and semantic information across hidden dimensions [10, 17]. The motif-constrained low-rank pathway provides a regularized interaction route among state dimensions, which may help organize hidden representations more effectively than an unconstrained low-rank coupling.

For long-sequence extrapolation, Motif-Mamba shows stronger stability when the test sequence length exceeds the training context. The induction heads task requires preserving cue–target associations over long temporal gaps, making it a controlled setting for evaluating long-context memory [37]. Standard Mamba benefits from selective state-space recurrence, but its mostly diagonal state evolution limits explicit communication among state channels [10]. The compact motif-guided pathway improves information propagation without dense attention.

Motif-2 achieves the best overall performance and belongs to the more stable group in the three-level motif hierarchy [12]. Compared with motifs containing richer recurrent feedback, Motif-2 provides a simpler local interaction pattern, which may better support memory preservation, controlled state propagation, and reduced interference. This interpretation highlights the organization of recurrent interactions as a potential design consideration beyond their overall strength. Its advantage over Motif-12 suggests that richer recurrent connectivity does not necessarily improve performance, although the ablation results do not establish stability as the cause.

Motif-Mamba introduces structural priors inspired by biological connectivity, using motif statistics as lightweight inspiration rather than a mechanistic model of neural circuits. Local organizational principles observed in biological connectivity may inspire more structured and interpretable mechanisms in neural architectures [38]. The motif prior makes part of the hidden-state evolution more analyzable, turning the recurrent update from a fully black-box process into a gray-box dynamical structure. Future work may extend this framework by assigning different motif distributions to different layers, tasks, or domains. Such priors could accommodate task-specific dynamical requirements, including stable memory preservation, flexible updating, and robust temporal decoding. Evaluating when these requirements favor different motif structures would help clarify the scope of the proposed approach.

Limitation. First, three-node motifs capture local connectivity but cannot fully represent higher-order interactions. Extending motif-constrained modeling to larger subgraphs, such as four-node motifs or hierarchical motif compositions, may provide a richer structural prior. Second, the current study mainly uses fixed representative motif types from different stability levels. Although this enables controlled analysis, mixed or adaptive motif priors may better balance stability and flexibility across tasks. Third, Motif-Mamba applies motif priors to state space dynamics, but how to define and compare motif-induced dynamics across different architectures remains an open question. Broader analysis may clarify how motif hierarchy relates to sequence-model behavior.

Acknowledgments

Funding: This work was supported by the Brain Science and Brain-like Intelligence Technology - National Science and Technology Major Project (2025ZD0217200), the Shanghai Municipal Science and Technology Project (Grant No. 25ZR1401370, 25LN3200400), Science and Technology Commission of Shanghai Municipality (25511102500), CAS Project for Young Scientists in Basic Research (YSBR-116), Youth Innovation Promotion Association CAS, Shanghai Leading Talent Program of Eastern Talent Plan, the Lingang Laboratory Fund (Grant No. LG-GG-202402-06-07, LGL-1987-09), Special Support Project of Guangdong Province (Grant No.0720240209). This work was also supported by Shanghai Maiqi Xinzhi Technology Co., Ltd. The numerical calculations in this study were carried out on the ORISE Supercomputer.

Author contributions: Tielin Zhang conceived the research topic. Chonghe Hao and Jian Zhang conducted the experiments. Yue Sun and Jian Zhang provided the theoretical contributions. Chonghe Hao and Jian Zhang wrote the manuscript. Yansong Wang provided research guidance. Wangzi Yao and Yunjie Yao contributed to the completion of the manuscript.

Competing interests: The authors declare that they have no competing interests.

References

[1] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

[2] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[4] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[5] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems*, 2020.

[6] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems*, 2022.

[7] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

[8] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, et al. Rwkv: Reinventing rnns for the transformer era. *arXiv preprint arXiv:2305.13048*, 2023.

[9] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.

[10] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[11] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[12] Zhang Jian, Sun Yue, Yao Wangzi, and Zhang Tielin. Functional building blocks of neural networks: From network motifs to collective dynamics. In *International Conference on Machine Learning*, 2026.

[13] Wangzi Yao, Yue Sun, Bo Xu, and Tielin Zhang. Critical-state-accelerated rnn-based reinforcement learning. *Neurocomputing*, 2026.

[14] Robert J Prill, Pablo A Iglesias, and Andre Levchenko. Dynamic properties of network motifs contribute to biological network organization. *PLoS biology*, 3(11):e343, 2005.

[15] Uri Alon. Network motifs: Theory and experimental approaches. *Nature Reviews Genetics*, 8 (6):450–461, 2007. doi: 10.1038/nrg2102.

[16] Tara N. Sainath, Brian Kingsbury, Vikas Sindhwani, Ebru Arisoy, and Bhuvana Ramabhadran. Low-rank matrix factorization for deep neural network training with high-dimensional output targets. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6655–6659, 2013.

[17] Francesca Mastrogiuseppe and Srdjan Ostojic. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, 99(3):609–623, 2018.

[18] Jimmy T. H. Smith, Andrew Warrington, and Scott W. Linderman. Simplified state space layers for sequence modeling. In *International Conference on Learning Representations*, 2023.

[19] Daniel Y. Fu, Tri Dao, Khaled K. Saab, Armin W. Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. In *International Conference on Learning Representations*, 2023.

[20] Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

[21] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[22] Seung Wook Oh, Julie A. Harris, Lydia Ng, Brent Winslow, Nicholas Cain, Stefan Mihalas, Quanxin Wang, Chris Lau, Leonard Kuan, Alex M. Henry, Marty T. Mortrud, Benjamin Ouellette, Thuc Nghi Nguyen, Staci A. Sorensen, Cliff R. Slaughterbeck, Wayne Wakeman, Yang Li, David Feng, Anh Ho, Eric Nicholas, Karla E. Hirokawa, Phillip Bohn, Kathryn M. Joines, Hanchuan Peng, Michael J. Hawrylycz, John W. Phillips, John G. Hohmann, Paul Wohnoutka, Charles R. Gerfen, Christof Koch, Amy Bernard, Chinh Dang, Allan R. Jones, and Hongkui Zeng. A mesoscale connectome of the mouse brain. *Nature*, 508(7495):207–214, 2014.

[23] Alexander J. Sadovsky and Jason N. MacLean. Mouse visual neocortex supports multiple stereotyped patterns of microcircuit activity. *Journal of Neuroscience*, 34(23):7769–7777, 2014.

[24] Federico Monti, Karl Otness, and Michael M. Bronstein. Motifnet: A motif-based graph convolutional network for directed graphs. *arXiv preprint arXiv:1802.01572*, 2018.

[25] Xuexin Chen, Ruichu Cai, Yuan Fang, Min Wu, Zijian Li, and Zhifeng Hao. Motif graph neural network. *IEEE Transactions on Neural Networks and Learning Systems*, 35(10):14833–14847, 2023.

[26] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9 (8):1735–1780, 1997.

[27] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. Language model evaluation harness. `https://github.com/EleutherAI/lm-evaluation-harness`, 2024.

[28] Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016.

[29] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[30] Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[31] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. In *arXiv preprint arXiv:1803.05457*, 2018.

[32] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. In *Communications of the ACM*, volume 64, pages 99–106, 2021.

[33] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: A suite for analyzing large language models across training and scaling. *arXiv preprint arXiv:2304.01373*, 2023.

[34] Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. Gpt-neo: Large scale autoregressive language modeling with mesh-tensorflow. *Zenodo*, 2021.

[35] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[36] Xuan Ma, Fabio Rizzoglio, Eric J. Perreault, Lee E. Miller, and Amy Kennedy. Using adversarial networks to extend brain computer interface decoding accuracy over time. *eLife*, 12:e84296, 2023. doi: 10.7554/eLife.84296.

[37] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

[38] Quanxin Wang, Song-Lin Ding, Yang Li, Josh Royall, David Feng, Phil Lesnar, Nile Graddis, Murtaza Naeemi, Ben Facer, Anh Ho, Tim Dolbeare, Bob Blanchard, Nick Dee, Wayne Wakeman, Karla E. Hirokawa, Aaron Szafer, Susan M. Sunkin, Seung Wook Oh, Amy Bernard, John W. Phillips, Michael Hawrylycz, Christof Koch, Hongkui Zeng, Julie A. Harris, and Lydia Ng. The allen mouse brain common coordinate framework: A 3d reference atlas. *Cell*, 181(4): 936–953.e20, 2020. doi: 10.1016/j.cell.2020.04.007.

A Notation

Table S1: Notation used throughout the paper.

Symbol	Definition
State space model	
x_t, $x(t)$	Discrete and continuous input.
h_t, $h(t)$	Discrete and continuous hidden state.
A_t, $A(t)$	Discrete and continuous state transition matrix.
B_t, $B(t)$	Discrete and continuous input projection matrix.
C_t	Output projection matrix.
$\bar{A}_t$, $\bar{B}_t$	Discretized transition and input terms.
Δ_t	Step size / selective timescale.
n, L	Hidden-state dimension and sequence length.
Low-rank motif pathway	
r	Low-rank dimension.
$M_I \in \mathbb{R}^{n\times r}$	Latent-to-state low-rank factor.
$M_J \in \mathbb{R}^{r\times n}$	State-to-latent low-rank factor.
$M = M_I M_J$	Low-rank recurrent coupling matrix.
$z_t = M_J h_t$	Low-dimensional latent state.
$v(z)$	Projected latent flow field.
k_{motif}	Motif coupling coefficient.
Δz_{norm}	Normalized latent-state displacement.
Motif regularization	
$\tilde{M}$	Differentiable adjacency proxy.
$\sigma(\cdot)$	Sigmoid function.
β, θ	Relaxation sharpness and threshold.
$\mathcal{M}(\cdot)$	Motif extraction operator.
$\mathbf{m}_M$	Learned motif frequency vector.
$\mathbf{m}_{\text{target}}$	Target motif frequency vector.
$m_{M,k}$, $m_{\text{target},k}$	Learned and target frequency of motif k.
K	Number of motif categories.
R^2	Motif-profile alignment score.
ϵ	Numerical stabilizer.
$\mathcal{L}_{\text{task}}$	Task loss.
$\mathcal{L}_{\text{motif}}$	Motif regularization loss.
$\mathcal{L}$	Total training objective.
λ	Motif regularization weight.
Motif counting	
N	Number of graph nodes.
i, j, k	Node indices.
$\widetilde{W}$	Binary adjacency matrix.
W, $\bar{W}$	Edge-present and edge-absent matrices.
P	Off-diagonal mask.
ℓ	All-one vector.
c_m	Count of the m-th directed three-node motif.
Proofs and experiments	
$\Gamma(t, \tau)$	Transition product from step τ to step t.
d_{state}, d_{conv}	Mamba state size and convolution width.
n_{layer}	Number of layers.
PPL, ACC, std	Perplexity, accuracy, and standard deviation.

B Proofs and Derivations

B.1 Derivation of Lemma 1

We consider the continuous-time dynamics of the Motif-Mamba system defined by the state equation

$$\dot{h}(\tau) = -A(\tau)h(\tau) + M_I M_J h(\tau) + B(\tau)x(\tau). \tag{13}$$

For a discretization interval $[\tau_{t-1}, \tau_t]$ with step size $\Delta_t = \tau_t - \tau_{t-1}$, the solution follows the variation of constants formula

$$\begin{aligned} h(\tau_t) = &\underbrace{\exp\left(-\int_{\tau_{t-1}}^{\tau_t} A(s)ds\right) h(\tau_{t-1})}_{\text{self-decay term}} \\ &+ \underbrace{\int_{\tau_{t-1}}^{\tau_t} \exp\left(-\int_{\tau}^{\tau_t} A(s)ds\right) \left(M_I M_J h(\tau) + B(\tau)x(\tau)\right) d\tau}_{\text{driving term}}. \end{aligned} \tag{14}$$

To derive the discrete-time update, we apply a piecewise constant approximation where the transition matrix $A(\tau) \approx A_t$ and the driving components $M_I M_J h(\tau) \approx M_I M_J h_{t-1}$ and $B(\tau)x(\tau) \approx B_t x_t$ remain constant over the interval. This assumption leads to the following integral approximation

$$\begin{aligned} h_t &\approx e^{-\Delta_t A_t} h_{t-1} + \left(\int_{\tau_{t-1}}^{\tau_t} e^{-(\tau_t - \tau)A_t} d\tau \right) (M_I M_J h_{t-1} + B_t x_t) \\ &= e^{-\Delta_t A_t} h_{t-1} + A_t^{-1} \left(I - e^{-\Delta_t A_t} \right) (M_I M_J h_{t-1} + B_t x_t) . \end{aligned} \tag{15}$$

By applying a first-order Taylor expansion $e^{-\Delta_t A_t} = I - \Delta_t A_t + O(\Delta_t^2)$ around $\Delta_t = 0$, the integral coefficient simplifies as follows

$$A_t^{-1} \left(I - e^{-\Delta_t A_t} \right) = A_t^{-1} \left(\Delta_t A_t + O(\Delta_t^2) \right) = \Delta_t I + O(\Delta_t^2) \approx \Delta_t I. \tag{16}$$

Substituting this approximation yields the discrete-time Motif-Mamba update without input derivative terms

$$h_t \approx e^{-\Delta_t A_t} h_{t-1} + \Delta_t \left(M_I M_J h_{t-1} + B_t x_t \right). \tag{17}$$

This update retains the exponential diagonal decay but is not the exact exponential-Euler step. For nonsingular A_t, the coefficient above equals $\Delta_t \varphi_1(-\Delta_t A_t)$, where $\varphi_1(Z) = \sum_{k=0}^{\infty} Z^k/(k+1)!$ also defines it for singular A_t. Freezing the low-rank driving state and replacing this coefficient by $\Delta_t I$ yield a local error of $O(\Delta_t^2)$ for bounded matrices, states, and inputs. The coefficient approximation affects both the low-rank and input terms, even when the input is constant within the interval. Thus the scheme is first order, not second order.

B.2 Proof of Lemma 2

Consider the discrete Motif-Mamba recurrence

$$h_t = e^{-\Delta_t A_t} h_{t-1} + \Delta_t M_I z_{t-1} + \Delta_t B_t x_t. \tag{18}$$

For $0 \le \tau \le t$, define the transition product

$$\Gamma(t, \tau) = e^{-\Delta_t A_t} e^{-\Delta_{t-1} A_{t-1}} \cdots e^{-\Delta_{\tau+1} A_{\tau+1}}, \tag{19}$$

with the convention $\Gamma(t, t) = I$. Equivalently, $\Gamma(t, \tau)$ satisfies the recursion $\Gamma(t, \tau) = \Gamma(t, t-1)\Gamma(t-1, \tau)$, where $\Gamma(t, t-1) = e^{-\Delta_t A_t}$. Using this notation, the one-step recurrence can be written in terms of Γ. Substituting the corresponding expression for h_{t-1} gives

$$\begin{aligned} h_t &= \Gamma(t, t-1) h_{t-1} + \Gamma(t, t) \left(\Delta_t M_I z_{t-1} + \Delta_t B_t x_t \right) \\ &= \Gamma(t, t-2) h_{t-2} + \Gamma(t, t-1) \left(\Delta_{t-1} M_I z_{t-2} + \Delta_{t-1} B_{t-1} x_{t-1} \right) \\ &\quad + \Gamma(t, t) \left(\Delta_t M_I z_{t-1} + \Delta_t B_t x_t \right). \end{aligned} \tag{20}$$

Repeating the same substitution from t back to 0 yields

$$h_t = \Gamma(t, 0) h_0 + \sum_{\tau=1}^{t} \Gamma(t, \tau) \left(\Delta_\tau M_I z_{\tau-1} + \Delta_\tau B_\tau x_\tau \right). \tag{21}$$

This proves the reconstruction formula. The expression shows that, once h_0, the transition parameters, the input history, and the low-rank coordinate history $\{z_{\tau-1}\}_{\tau=1}^{t}$ are given, the hidden state h_t is determined by the recurrence.

C Calculating the Frequency of Network Motifs

Let $\widetilde{W} \in \{0, 1\}^{N \times N}$ be the adjacency matrix, where $\widetilde{W}_{jk} = 1$ indicates a directed connection from node k to node j, and $\widetilde{W}_{jk} = 0$ indicates its absence. Let $L = \mathbf{1}_{N,1}$ be an all-one vector. For any matrix $X \in \mathbb{R}^{N \times N}$,

$$L^\top X L = \sum_{i,j} X_{ij}. \tag{22}$$

To exclude self-loops, define

$$P = \mathbf{1}_{N \times N} - I_{N \times N}. \tag{23}$$

Then

$$W_{ji} = \widetilde{W}_{ji} P_{ji}, \qquad \bar{W}_{ji} = (P - \widetilde{W})_{ji}, \tag{24}$$

where W_{ji} denotes the presence of a directed edge and $\bar{W}_{ji}$ denotes its absence.

For motif2, we consider the in-star pattern where two nodes j and k both project to node i, while there are no reverse edges and no direct edges between j and k. Its motif count is

$$c_2(W) = \frac{1}{2} \sum_{i,j,k} W_{ij} \bar{W}_{ji} W_{ik} \bar{W}_{ki} \bar{W}_{kj} \bar{W}_{jk}. \tag{25}$$

Equivalently, using matrix multiplication and Hadamard products, this count can be written as

$$\begin{aligned}
c_2(W) &= \frac{1}{2} \sum_{i,j,k} W_{ij} \bar{W}_{ji} W_{ik} \bar{W}_{ki} \bar{W}_{kj} \bar{W}_{jk} \\
&= \frac{1}{2} \sum_{i,j,k} \widetilde{W}_{ij} (P - \widetilde{W})_{ji} \widetilde{W}_{ik} (P - \widetilde{W})_{ki} (P - \widetilde{W})_{kj} (P - \widetilde{W})_{jk} \\
&= \frac{1}{2} \sum_{i,j,k} \widetilde{W}_{ji}^{\top} (P - \widetilde{W})_{ji} \widetilde{W}_{ik} (P - \widetilde{W})_{ik}^{\top} (P - \widetilde{W})_{jk}^{\top} (P - \widetilde{W})_{jk} \\
&= \frac{1}{2} \sum_{i,j,k} \left[\widetilde{W}^{\top} \odot (P - \widetilde{W}) \right]_{ji} \left[\widetilde{W} \odot (P - \widetilde{W})^{\top} \right]_{ik} \left[(P - \widetilde{W})^{\top} \odot (P - \widetilde{W}) \right]_{jk} \\
&= \frac{1}{2} \sum_{j,k} \left(\sum_{i} \left[\widetilde{W}^{\top} \odot (P - \widetilde{W}) \right]_{ji} \left[\widetilde{W} \odot (P - \widetilde{W})^{\top} \right]_{ik} \right) \left[(P - \widetilde{W})^{\top} \odot (P - \widetilde{W}) \right]_{jk} \\
&= \frac{1}{2} \sum_{j,k} \left[\left(\widetilde{W}^{\top} \odot (P - \widetilde{W}) \right) \left(\widetilde{W} \odot (P - \widetilde{W})^{\top} \right) \right]_{jk} \left[(P - \widetilde{W})^{\top} \odot (P - \widetilde{W}) \right]_{jk} \\
&= \frac{1}{2} \sum_{j,k} \left[\left(\widetilde{W}^{\top} \odot (P - \widetilde{W}) \right) \left(\widetilde{W} \odot (P - \widetilde{W})^{\top} \right) \odot (P - \widetilde{W})^{\top} \odot (P - \widetilde{W}) \right]_{jk} \\
&= \frac{1}{2} L^{\top} \left[\left(\widetilde{W}^{\top} \odot (P - \widetilde{W}) \right) \left(\widetilde{W} \odot (P - \widetilde{W})^{\top} \right) \odot (P - \widetilde{W})^{\top} \odot (P - \widetilde{W}) \right] L.
\end{aligned} \tag{26}$$

The counts of the other directed three-node motifs can be obtained analogously by replacing the corresponding edge-present and edge-absent terms according to their motif topologies.

D Additional Experiments

D.1 Minimum Rank for Applying Motif Constraints

We further analyze the minimum rank required for representing the motif constraint. For each three-node motif, its adjacency pattern can be written as a motif matrix

$$M = M_I M_J, \tag{27}$$

where

$$M_I \in \mathbb{R}^{n \times r}, \qquad M_J \in \mathbb{R}^{r \times n}. \tag{28}$$

Since

$$\operatorname{rank}(M) \leq r, \tag{29}$$

the rank r determines the maximum structural complexity that can be represented by the low-rank motif factorization.

As shown in Figure S1, although the motif matrix is defined over three nodes, most motif patterns do not require rank three. In practice, rank-one factorization is already sufficient to represent or constrain most motif structures. The main exceptions are motif 3 and motif 10. Motif 3 corresponds to a feed-forward directed structure, while motif 10 corresponds to a cyclic directed structure. These two structures require richer directional interactions than those expressible by a rank-one motif matrix.

When the rank is increased to 2, almost all motif matrices can be effectively represented by the factorization $M = M_I M_J$. This observation shows that the motif constraint can be imposed with a very small rank. Therefore, Motif-Mamba can incorporate biologically inspired motif priors without relying on a high-rank or dense coupling matrix.

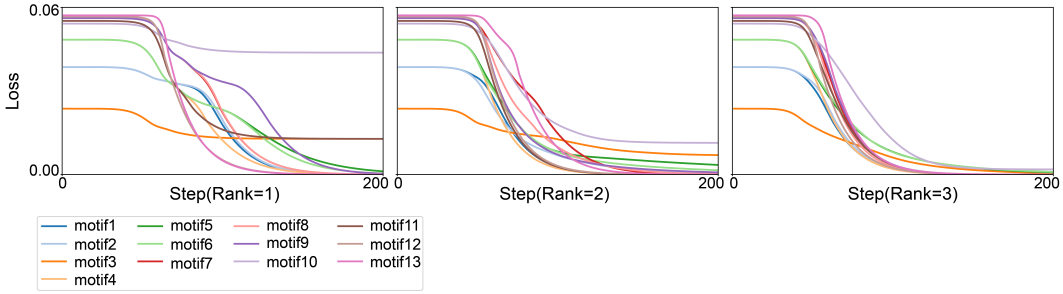


Figure S1: Rank requirement for low-rank motif factorization. Each three-node motif matrix is factorized as $M = M_I M_J$, where $M_I \in \mathbb{R}^{3\times r}$ and $M_J \in \mathbb{R}^{r\times 3}$. Rank-one factorization is sufficient for most motif constraints, except motif 3 and motif 10. Motif 3 represents a feed-forward directed structure, while motif 10 represents a cyclic directed structure. Increasing the rank to two can effectively support almost all motif constraints considered in this work.

D.2 Additional Parameter Cost

Motif-Mamba introduces only a small number of additional parameters compared with the original Mamba architecture. The additional parameters mainly come from the motif-constrained low-rank coupling module, including the low-rank motif factors M_I and M_J, together with a learnable scalar coefficient k_{motif}. Instead of assigning an independent motif coupling matrix to every channel, Motif-Mamba adopts a block-wise sharing strategy.

Specifically, different Mamba blocks use different motif factorization matrices M_I and M_J, allowing each block to learn its own motif-guided recurrent structure. This design provides sufficient flexibility across network depth, since different layers can capture different levels of temporal and structural interactions. However, within the same Mamba block, all channels share the same pair of motif matrices M_I and M_J. The scalar coefficient k_{motif} is also assigned at the block level to control the strength of the low-rank motif coupling. Therefore, the motif prior is applied consistently across channels in the same block, rather than introducing a separate set of motif parameters for each channel.

For each Mamba block, the additional number of parameters is

$$2d_{\text{state}}r + 1, \tag{30}$$

where $2d_{\text{state}}r$ corresponds to the two low-rank motif factors $M_I \in \mathbb{R}^{d_{\text{state}}\times r}$ and $M_J \in \mathbb{R}^{r\times d_{\text{state}}}$, and the additional 1 corresponds to the learnable scalar coefficient k_{motif}. Thus, for a model with n_{layer} blocks, the total number of additional parameters is

$$n_{\text{layer}}\left(2d_{\text{state}}r + 1\right). \tag{31}$$

This block-wise parameterization greatly reduces the additional parameter cost. Because M_I and M_J are shared among channels within each block, the number of newly introduced parameters mainly depends on the number of blocks and the selected low-rank dimension, rather than scaling independently with every channel. As a result, Motif-Mamba can introduce biologically inspired motif constraints into the recurrent dynamics while preserving nearly the same model scale as the original Mamba.

An additional advantage of this design is its extensibility. Since different blocks maintain independent motif factorization matrices, different motif priors can be assigned to different blocks in future extensions. For example, lower blocks may be constrained by simple local interaction motifs, while higher blocks may incorporate more complex recurrent or cyclic motifs. This provides a flexible way to study hierarchical motif organization across depth, enabling Motif-Mamba to serve as a general framework for injecting layer-specific biological structural priors into sequence models.

Table S2: Additional parameter cost of Motif-Mamba. The added parameters come from the low-rank motif factors M_I and M_J, together with a learnable scalar coefficient k_{motif}. Different blocks use different motif factorization matrices, while channels within the same block share the same factors.

Model	Original Params	n_{layer}	d_{state}	Rank r	Added Params	Ratio
Motif-Mamba-130M	129,135,360	24	16	2	1,560	0.001208%
Motif-Mamba-370M	371,516,416	48	16	2	3,120	0.000840%
Motif-Mamba-790M	793,204,224	48	16	2	3,120	0.000393%
Motif-Mamba-1.4B	1,372,178,432	48	16	2	3,120	0.000227%

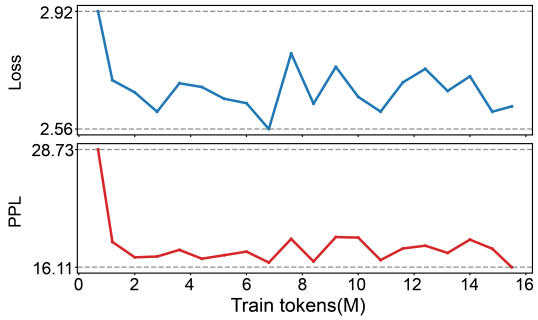


Figure S2: Training dynamics of Motif-Mamba with respect to the number of training tokens. The continued training data are randomly sampled from the Pile dataset, following the same data source used for the original Mamba backbone. The loss and perplexity decrease quickly in the early stage and become relatively stable after about 2M tokens. Continued training of the original Mamba backbone under the same token budget does not lead to noticeable improvement, indicating that the performance gain mainly comes from the motif-constrained low-rank recurrent pathway. Based on this trend, we train Motif-Mamba with 10M tokens in the main experiments.

As shown in Table S2, the parameter sizes of Motif-Mamba remain very close to those of the corresponding Mamba baselines across different model scales. This comparison indicates that the performance improvement of Motif-Mamba is not mainly caused by increasing the number of parameters. Instead, the gain comes from introducing motif-guided structural priors through a lightweight block-wise low-rank coupling design.

D.3 Train Tokens

We further analyze the training dynamics of Motif-Mamba with respect to the number of training tokens. The continued training data are randomly sampled from the Pile dataset, which is the same data source used for training the original Mamba models. This setting keeps the data distribution consistent with the Mamba backbone and avoids introducing additional gains from a different training corpus.

Figure S2 shows the loss and perplexity curves during training with up to 16M tokens. Both loss and PPL decrease rapidly at the early stage of training and become largely stable after approximately 2M tokens. Although small fluctuations remain in the later training stage, the overall performance does not show a clear additional improvement with more tokens.

To further exclude the possibility that the observed improvement is mainly caused by continued training, we also continue training the original Mamba backbone under the same token budget and training setting. The continued training of Mamba does not bring noticeable performance improvement, suggesting that the gains of Motif-Mamba mainly come from the motif-constrained low-rank recurrent pathway rather than from additional training tokens alone.

Based on this observation, we use 10M training tokens for the main experiments. This setting provides a stable training regime while avoiding unnecessary computational cost from longer training.

D.4 Evaluation of All Thirteen Motifs

We evaluate all thirteen directed three-node motif variants using the same Mamba-130M backbone, low-rank dimension, parameter budget, training data, and additional 10M-token continued-pretraining budget. Table S3 reports the complete results. Motif-2 achieves the highest reported average accuracy (45.1%), followed by Motif-1 (45.0%). This ranking is consistent with the stability-oriented motif selection, but does not establish a causal relationship between stability and task performance.

Table S3: Complete thirteen-motif comparison on the 130M backbone (PPL and accuracy, %). Mamba uses the matched 10M-token continued-pretraining baseline.

Model	LAMB. PPL ↓	LAMB. ACC ↑	HellaS. ACC ↑	PIQA ACC ↑	Arc-E ACC ↑	Arc-C ACC ↑	WinoGr. ACC ↑	Average ACC ↑
Mamba	16.07	44.3	35.3	64.5	48.0	24.3	51.9	44.7
Motif-1-Mamba	15.53	44.6	35.5	64.7	48.2	24.4	52.3	45.0
Motif-2-Mamba	15.47	44.7	35.7	64.7	48.5	24.6	52.4	45.1
Motif-3-Mamba	15.60	44.4	35.3	64.3	47.7	24.1	52.3	44.7
Motif-4-Mamba	16.22	44.1	35.3	63.6	48.2	24.3	52.4	44.7
Motif-5-Mamba	15.59	44.4	35.3	64.3	48.8	24.4	52.0	44.9
Motif-6-Mamba	15.96	44.2	35.1	64.4	47.8	24.4	52.0	44.7
Motif-7-Mamba	15.71	43.8	35.1	64.5	48.2	24.5	51.9	44.7
Motif-8-Mamba	15.92	44.2	35.2	64.6	48.2	24.3	52.1	44.8
Motif-9-Mamba	16.41	43.9	35.1	63.9	47.9	24.5	52.4	44.6
Motif-10-Mamba	15.75	44.4	35.2	64.4	48.4	24.3	52.5	44.9
Motif-11-Mamba	16.05	44.0	35.1	63.8	48.1	24.3	52.5	44.6
Motif-12-Mamba	15.88	44.2	35.5	64.6	48.4	24.5	51.9	44.9
Motif-13-Mamba	16.19	44.0	35.4	64.4	48.4	24.2	51.4	44.6

D.5 Empirical Throughput and Latency

We compare Mamba and Motif-Mamba under matched model scale, hardware, numerical precision, batch size, and sequence length. Throughput is measured in tokens per second, and latency measures one full-sequence forward pass. Table S4 reports the measurements and their variability. Across the four tested lengths, the unweighted mean throughput reduction is 2.89% and the mean latency increase is 3.03%. At length 8192, throughput decreases by 5.85% and latency increases by 6.21%. These measurements characterize full-sequence forward execution, rather than autoregressive decoding or training throughput.

Table S4: Full-sequence forward throughput and latency. Relative changes are measured against Mamba.

Length	Mamba tok/s	Motif tok/s	Δ tok/s (%)	Mamba ms	Motif ms	Δ ms (%)
1024	43835.09 ± 276.62	43653.18 ± 201.99	-0.41	23.36 ± 0.15	23.46 ± 0.11	$+0.42$
2048	86736.62 ± 307.66	85984.23 ± 242.88	-0.87	23.61 ± 0.08	23.82 ± 0.07	$+0.88$
4096	110734.37 ± 293.82	105848.10 ± 290.18	-4.41	36.99 ± 0.10	38.70 ± 0.11	$+4.62$
8192	130242.33 ± 702.54	122628.55 ± 412.81	-5.85	62.90 ± 0.34	66.80 ± 0.22	$+6.21$

E Experimental Details

E.1 Model Variants

We evaluate several variants of Mamba and Motif-Mamba. Standard Mamba is used as the main baseline. Motif-Mamba extends Mamba by adding a low-rank recurrent coupling term into the state update. To analyze the role of motif constraints, we further evaluate Motif-Blank-Mamba, which keeps the same low-rank pathway but removes the motif constraint, as well as Motif-2-Mamba and Motif-12-Mamba, which apply different motif-based structural constraints.

Unless otherwise specified, Motif-Mamba refers to the Motif-2-constrained variant in the main experiments. All Motif-Mamba variants are built on the same Mamba backbone as their corresponding baselines, and the remaining architecture is kept unchanged except for the inserted low-rank matrices M_I and M_J.

E.2 Training Setup for Language Evaluation and Motif Ablation

The training setup in this subsection is used for the language model evaluation and motif ablation experiments. For these experiments, Motif-Mamba is initialized from pretrained Mamba checkpoints and then continuously trained after inserting the low-rank recurrent coupling matrices M_I and M_J. This setting allows us to evaluate whether the proposed motif-constrained dynamics can improve the original Mamba backbone without training the whole model from scratch.

All models are trained on NVIDIA A800 GPUs. The context length is fixed to 2048 tokens. We use 10M additional training tokens for the main language evaluation and motif ablation experiments to balance training stability and computational cost.

Table S5 summarizes the main hyperparameters used for continued training at different model scales. Motif-Mamba models are initialized from pretrained Mamba checkpoints and continuously trained with the inserted low-rank recurrent coupling.

Table S5: Training hyperparameters for language model evaluation and motif ablation.

Model	GPUs	Train Tokens	Batch Size	Learning Rate
Motif-Mamba-130M	1×A800	10M	2	1e-3
Motif-Mamba-370M	1×A800	10M	1	1e-3
Motif-Mamba-790M	2×A800	10M	1	1e-3
Motif-Mamba-1.4B	4×A800	10M	1	1e-3

E.3 Long-Sequence Extrapolation

The long-sequence extrapolation experiment uses a separate training setup from the language model evaluation experiments. In this experiment, models are trained on the induction heads task with a sequence length of 256 and a vocabulary size of 16. The goal is to evaluate whether each model can generalize to contexts longer than those observed during training.

At test time, models are evaluated on sequence lengths ranging from 64 to 16384 tokens. We report next-token prediction accuracy on the induction target positions and visualize performance as the test sequence length increases. All models in this experiment use the same data generation process, optimizer, batch size, learning rate schedule, and number of training steps.

Table S6 summarizes the model configurations used in the long-sequence extrapolation experiment. All models are kept at a comparable scale to ensure a fair comparison across different sequence modeling architectures.

Table S6: Model configurations used in the long-sequence extrapolation experiment.

Model	#Layers	Hidden Dim.	Additional Settings
Transformer	2	64	n_heads = 4, ffn_mult = 2.0
LSTM	2	64	batch_first = True, no heads
RWKV	2	72	batch_first = True, no heads
Mamba	2	64	$d_{\text{state}} = 16$, $d_{\text{conv}} = 4$, expand = 2
Motif-Mamba	2	64	$d_{\text{state}} = 16$, $d_{\text{conv}} = 4$, expand = 2, rank = 2

As a complementary quantitative summary, Table S7 reports the accuracy values across different evaluation lengths. Motif-Mamba maintains higher accuracy than the compared baselines as the test sequence length increases, indicating stronger extrapolation ability beyond the training context.

E.4 BCI Decoding

For the BCI decoding experiment, we use a separate set of model configurations from the language modeling experiments. The goal is to compare different sequence models under a controlled neural decoding setting. All models are trained and evaluated using the same data split, preprocessing pipeline, and evaluation protocol.

Table S7: Accuracy on the induction heads extrapolation task (%). Models are trained with sequence length 256 and evaluated on different test sequence lengths.

Model	64	128	256	512	1024	2048	4096	8192	16384
LSTM	99.12	98.74	97.86	28.63	8.27	3.14	1.78	1.83	1.76
Transformer	99.43	99.21	98.67	34.58	10.36	4.82	2.87	2.91	2.76
RWKV	99.76	99.48	98.91	67.42	41.73	23.68	15.42	11.36	7.84
Mamba	99.82	99.61	99.08	75.64	48.72	30.86	20.74	15.28	10.93
Motif-Mamba	99.96	99.84	99.37	81.72	55.63	35.81	23.76	17.42	11.87

The model input is a multi-channel neural activity sequence, and the target is the corresponding two-dimensional wrist force/kinematic trajectory. We use mean squared error and coefficient of determination as decoding metrics. For single-day evaluation, each recording day is split into training, validation, and test segments. For cross-day generalization, models are trained on sessions before the dashed line and evaluated on held-out future sessions. Error bars are computed across repeated runs with different random seeds.

Table S8 summarizes the model configurations used in the BCI decoding experiment. LSTM and Transformer are included as common sequence modeling baselines. Mamba and Motif-Mamba share the same backbone configuration, while Motif-Mamba additionally introduces motif-constrained low-rank recurrent coupling with rank $r = 2$.

Table S8: Model configurations used in the BCI decoding experiments. Motif-Mamba and Mamba use the same backbone configuration.

Model	#Layers	Hidden Dim.	Additional Settings
LSTM	2	64	dropout = 0.2
Transformer	2	64	n_heads = 4, dim_ff = 128, dropout = 0.2
Mamba	2	56	$d_{\text{state}} = 8$, $d_{\text{conv}} = 4$, expand = 1
Motif-Mamba	2	56	$d_{\text{state}} = 8$, $d_{\text{conv}} = 4$, expand = 1, rank = 2

F Code Availability

The code repository is `https://github.com/4354ce38/Motif-Mamba`.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction summarize the proposed Motif-Mamba framework, the motif-constrained low-rank coupling mechanism, and the evaluated tasks, including long-sequence modeling and BCI decoding. The claims are supported by the experimental results and ablation studies reported in the paper, and we avoid making claims beyond the evaluated scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Limitation paragraph in Section 5 discusses the restricted expressiveness of three-node motifs, the use of fixed representative motif types, and the open question of comparing motif-induced dynamics across architectures. The Discussion also notes that the ablation results do not establish stability as the cause of the observed performance differences.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper includes two lemmas related to the exponential-Euler approximation and the reconstruction property of the low-rank recurrent pathway. The corresponding assumptions and derivations are provided in Appendices B.1 and B.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the main information needed to reproduce the experimental results, including the model architecture, motif-constrained low-rank coupling design, datasets, training settings, evaluation protocols, baseline models, and metrics. Implementation details and additional experimental settings are provided in the main text and appendix to support reproducibility of the main claims and conclusions.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The public repository at `https://github.com/4354ce38/Motif-Mamba`, linked in Appendix F, provides the model implementation and training and evaluation scripts for language modeling and induction tasks. However, large datasets and checkpoints are not included, and complete data-preparation instructions and end-to-end reproduction procedures covering all experiments, including BCI decoding, are not provided. We therefore answer No to the full requirement.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the training and test details needed to understand the results, including dataset splits, model configurations, optimizer type, learning rate, batch size, training length, evaluation metrics, baselines, and ablation settings. Hyperparameters are reported in the experimental setup and appendix, and were selected based on validation performance or standard settings used by the corresponding baselines.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Figure 4 reports standard deviations for BCI decoding, with $n = 5$ stated for cross-day evaluation; Appendix E identifies different random seeds as the source of repeated-run variability. Table S4 also reports variability, but its statistic and repetition count are not defined. The language model and motif-ablation tables do not report uncertainty estimates, so statistical reporting is incomplete across the main experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the main computational resources required to reproduce the experiments, including the type of GPUs used and relevant implementation settings. Resource information is provided for the main experiments and additional details are included in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research conducted in this paper conforms to the NeurIPS Code of Ethics. The work focuses on model development and evaluation using standard machine learning benchmarks and appropriately handled neural decoding datasets. We do not use private or personally identifiable information, and all experiments are conducted in accordance with applicable data-use policies and ethical guidelines.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: The current version does not include a complete discussion of both positive and negative societal impacts. We will add a dedicated discussion covering potential benefits, such as efficient long-sequence modeling and improved BCI decoding, as well as potential risks related to misuse, privacy, fairness, and responsible deployment.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release data or models with a high risk for misuse, such as large-scale pre-trained language models, image generators, or scraped datasets. The work focuses on architectural modifications and empirical evaluation. Therefore, safeguards for high-risk data or model release are not applicable.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper credits the original creators of the assets used in this work, including Mamba, benchmark datasets, baseline models, and evaluation tools. We cite the corresponding publications and repositories where applicable. The use of code, data, and models follows their licenses and terms of use, and any dataset-specific access conditions are respected.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The released Motif-Mamba implementation is a new code asset, available through Appendix F. The repository includes a README describing its contents and local-path configuration notes, together with an Apache-2.0 license file. However, documentation of the complete environment, data preparation, and reproduction commands for all reported experiments remains incomplete; we therefore answer No rather than NA.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or human-subject research. The BCI decoding experiments use existing non-human primate neural datasets, so participant instructions, screenshots, and compensation details are not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not conduct human-subject studies or collect new human participant data. The BCI decoding experiments use existing non-human primate neural datasets under their corresponding access and usage conditions. Therefore, human participant risk disclosure and IRB approval for human-subject research are not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The paper does not use LLMs as an important, original, or non-standard component of the core research methodology. Any use of LLMs, if applicable, was limited to writing, editing, or formatting assistance and did not affect the proposed method, experimental design, scientific conclusions, or originality of the work.

    Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.