Emergence of the Primacy Effect in Structured State-Space Models

Takashi Morita TMORITA@ALUM.MIT.EDU
Academy of Emerging Sciences, Chubu University, Japan
Institute for Advanced Research, Nagoya University, Japan

Editors: Hung-yi Lee and Tongliang Liu

Abstract

Structured state-space models (SSMs) have been developed to offer more persistent memory retention than traditional recurrent neural networks, while maintaining real-time inference capabilities and addressing the time-complexity limitations of Transformers. Despite this intended persistence, the memory mechanism of canonical SSMs is theoretically designed to decay monotonically over time, meaning that more recent inputs are expected to be retained more accurately than earlier ones. Contrary to this theoretical expectation, however, the present study reveals a counterintuitive finding: when trained and evaluated on a synthetic, statistically balanced memorization task, SSMs predominantly preserve the *initially* presented data in memory. This pattern of memory bias, known as the *primacy effect* in psychology, presents a non-trivial challenge to the current theoretical understanding of SSMs and opens new avenues for future research.

Keywords: primacy effect; state space models; long-term memory; memory bias

1. Introduction

In recent years, *structured state-space models* (SSMs) have garnered increasing attention as a backbone of next-generation artificial intelligent systems (Gu et al., 2022b; Gu and Dao, 2024). SSMs were developed to provide more persistent memory retention than traditional recurrent neural networks (RNNs), while maintaining real-time inference capabilities and addressing the time-complexity limitations of Transformers.

Despite this intended persistence, the memory mechanism of canonical SSMs is theoretically designed to decay monotonically over time (Gu et al., 2023); that is, more recent inputs are expected to be retained more accurately than earlier ones. For example, when a sequence of random integers such as $49, 75, \ldots, 5, 38$ is presented in that order, the final items $(5, 38)$ are theoretically more likely to be recalled accurately at the end of the sequence (blue curve in Figure 1).

Contrary to this theoretical expectation, however, the present study reveals a counterintuitive finding: when trained and evaluated on a synthetic, statistically balanced memorization task, SSMs predominantly preserve the *initially* presented data (e.g., $49, 75$) in memory (orange curve in Figure 1). This memory bias is known as the *primacy effect* in psychology; human and animal memory for sequentially presented items tends to be more accurate for those appearing at the beginning of the sequence (Ebbinghaus, 1913; Murdock, 1962; Glanzer and Cunitz, 1966).[1] This finding presents a non-trivial puzzle for existing theories and opens new avenues for research on SSMs.

1. Human and animal memory is also known to be more accurate for the most recently observed items—a phenomenon termed the *recency effect*. The theoretical design of SSMs aligns with this opposite pattern.

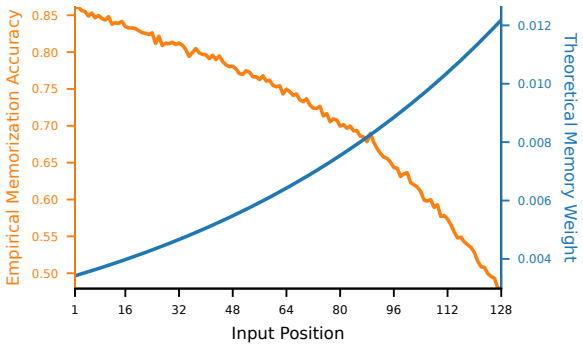

Figure 1: Graphical abstract of the key finding. When trained on a memorization task with sequentially presented items, the SSM achieves the highest accuracy for the earliest items (orange curve). This contrasts with the theoretical design of the SSM's memory mechanism, which assigns exponentially diminishing importance to older observations (blue curve).

The remainder of this paper is organized as follows. §2 first reviews the formalization and memory characteristics of SSMs. The section also discusses data-driven primacy effects observed in large language models—which inherit human biases embedded in linguistic data—and contrasts these findings with the statistically balanced setting examined in this study. §3 then details the task and model specifications, followed by results in §4. Finally, §5 summarizes the findings and discusses their implications in the context of prior work.

2. Preliminaries and Related Studies

2.1. Structured State-Space Models

2.1.1. Formalization

To achieve more persistent memory retention than traditional RNNs, the SSM leverages continuous-time dynamics, replacing the discrete-time framework of RNNs (Zhang et al., 2018; Voelker et al., 2019; Gu et al., 2020). The fundamental principle underlying these models is the polynomial approximation of observed signals (called the *HiPPO* framework; Gu et al., 2020). Specifically, given a *single*-channel input signal $x(\cdot)$, represented as a function of time, its approximation up to a given time point t can be approximated by a linear combination of polynomials:

$$x|_{\leq t}(\cdot) \approx \sum_{n=0}^{N-1} h_n(t) P_n(\cdot) \tag{1}$$

where P_n denotes the basis polynomial of degree n and $\{h_n(t)\}_{n=0}^{N-1}$ are the optimal coefficients for the approximation at time t. When these polynomials form an orthogonal basis with respect to a time-dependent measure $d\mu^{(t)}(\cdot)$, the optimal coefficients can be

determined as:

$$h_n(t) = \langle x|_{\leq t}, P_n \rangle = \int x|_{\leq t}(s) P_n(s)\, d\mu^{(t)}(s) \tag{2}$$

Then, these coefficients offer a finite- and constant-dimensional representation of the input signal up to time t. The framework can be naturally extended to multi-channel signals by performing channel-wise approximations in parallel, yielding $h_n^{(m)}(t)$ for each channel m.

For the polynomial coefficients to serve as a "memory" of the input signal, their temporal evolution must be trackable in an *online* manner; that is, the polynomial approximation at time t should not refer back to past values of the input signal, $x|_{\leq t}(s)$ $(0 \leq s < t)$. Fortunately, for certain families of polynomials, including Legendre, Laguerre, and Fourier basis,[2] the coefficient dynamics can be described by an ordinary differential equation (ODE; Gu et al., 2020):

$$\frac{d}{dt}\mathbf{h}(t) = A\mathbf{h}(t) + Bx(t) \tag{3}$$

where $\mathbf{h}(t) := (h_0(t), \ldots, h_{N-1}(t))^\mathsf{T}$, and A and B—referred to as the *state* and *input* matrices, respectively—are of size $N \times N$ and $N \times 1$. The values on A and B depend on the choice of the underlying polynomial basis, and can also be adjusted via gradient-based optimization. A feedforward transformation of the state vector $\mathbf{h}(t)$ (achieved via left-multiplication by another matrix C) yields a (possibly multi-channel) output signal $\mathbf{y}(t) = C\mathbf{h}(t) \in \mathbb{R}^M$.[3] The resulting mapping $x \mapsto \mathbf{y}$ defines the SSM (Gu et al., 2021, 2022b).

In practice, continuous-time recordings of an input signal $x(t)$ are not available; instead, empirical data consist of discrete-time samples at $t = t_1, \ldots, t_L$. Consequently, the SSM matrices must also be discretized in order to convert the ODE in Eq. 3 to a discrete recurrent system, analogous to RNNs:

$$\mathbf{h}(t_j) = \bar{A}\mathbf{h}(t_{j-1}) + \bar{B}x(t_j) \tag{4}$$

where $\bar{A}$ and $\bar{B}$ represent the discretized versions of A and B, respectively. A commonly used discretization technique is the bilinear method (Tustin, 1947), which yields:

$$\bar{A} := \left(I - \frac{\Delta t}{2}A\right)^{-1}\left(I + \frac{\Delta t}{2}A\right) \qquad \bar{B} := \left(I - \frac{\Delta t}{2}A\right)^{-1}\Delta t B \tag{5}$$

where $\Delta t := t_{j+1} - t_j$ $(\forall j = 1, \ldots, L-1)$ defines the time-step size. This time-step parameter is treated as learnable, allowing the model to automatically adjust the time scale of its state-space dynamics to align with that of the input signal.[4] Moreover, in multi-channel settings,

2. The Fourier approximation (or transform) is not based on polynomials, but the theory can be generalized to incorporate it by taking the complex-valued basis $z^n := e^{2\pi i n s}$ and a measure on the unit circle (Gu et al., 2020).
3. The general formulation of the SSM incorporates an additional matrix D, which establishes a direct feedforward connection between the input and output signals, expressed as $\mathbf{y}(t) = C\mathbf{h}(t) + D\mathbf{x}(t)$. In practice, however, D is often set to the identity matrix, effectively reducing the feedforward transformation to a simple residual connection (He et al., 2016).
4. Despite the data-driven adjustability of Δt, it is important to recognize that discretization methods are generally designed under the assumption of sufficiently small values on this parameter, for effectively

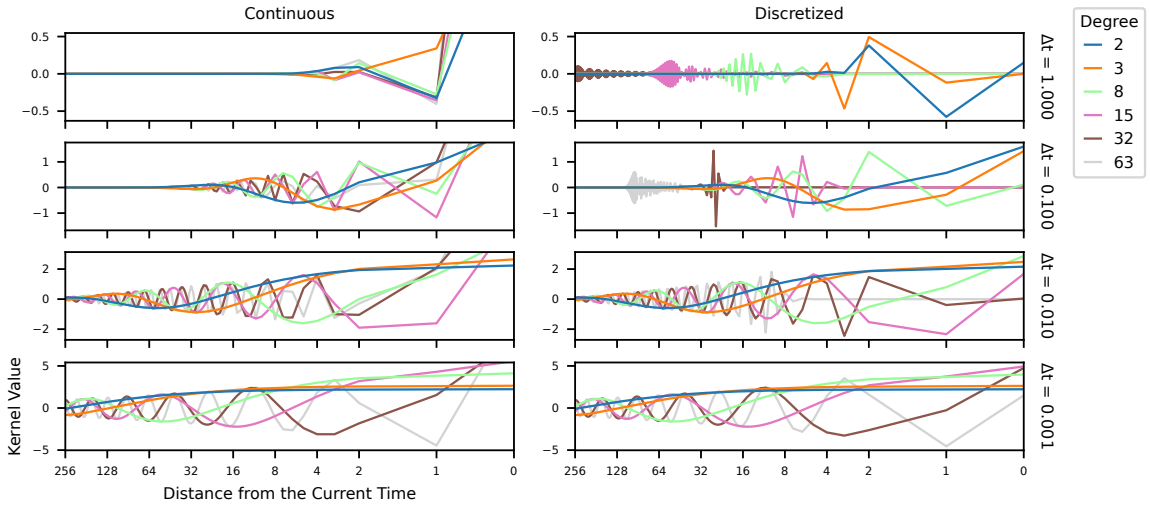

Figure 2: Illustration of the continuous (left; $K(\tau) := e^{\tau A}B$) and discretized (right; $\bar{K}_j := \bar{A}^j\bar{B}$) SSM kernels based on Legendre polynomials with an exponentially decaying measure (Gu et al., 2023). To aid intuitive understanding, the horizontal axis has been flipped, so that kernel values multiplied with past inputs appear on the left (unlike in the standard visualization of convolutional kernels, where they are placed on the right). The distinct line colors represent selected entries of the kernel vectors, $K_N(\tau)$ and $\bar{K}_{j,n}$, each corresponding to the approximating polynomial of degree $n \in \{2, 3, 8, 15, 32, 63\}$. The kernels were evaluated at $\tau = j\Delta t$ for $j = 0, \ldots, 255$ and $\Delta t \in \{0.001, 0.01, 0.1, 1.0\}$. Increasing Δt results in growing discrepancies between the continuous and discretized kernels.

the model can represent multi-scale dynamics by assigning distinct $\Delta t^{(m)}$ values to each channel m.

This flexibility extends the applicability of SSMs to inherently discrete data lacking overt continuous dynamics (e.g., text languages; Gu and Dao, 2024; Dao and Gu, 2024); the model can jointly lean latent representations (or embeddings) of discrete inputs along with their (pseudo-)continuous dynamics via gradient-based optimization. Furthermore, SSMs can be hierarchically stacked to construct deeper and more expressive models, where each layer processes latent signals received from lower layers.

2.1.2. Prior Studies on the Memory Dynamics of SSMs

Previous studies have identified that the time-step size, Δt, as a critical factor in determining the success/failure of SSMs. Intuitively, a small Δt results in minor state updates, yielding slow dynamics in the state space, $\mathbf{h}(t_{j+1}) - \mathbf{h}(t_j)$; conversely, a large Δt induces rapid state transitions (Gu et al., 2021; Gu and Dao, 2024). As a consequence, Δt governs the

approximating the limit $\Delta t \to 0$. Consequently, setting Δt too large introduces discrepancies between the continuous and discretized dynamics (as illustrated by the contrast between the left vs. right panels in Figure 2).

memory decay properties of the SSM; although the models assume an exponentially decaying measure in continuous time (particularly when employing Legendre/Laguerre polynomials; Gu et al., 2020, 2023), choosing small Δt values can allocate relatively large weights to input samples from distant time steps, thereby compromising single-step discriminability, which is better preserved with larger Δt (Figure 2). Notably, when an SSM employs a measure with fixed-length support—such as that used in the Fourier basis—Δt^{-1} corresponds to the model's effective memory length (Gu et al., 2023).

Owing to the rich theoretical foundations, previous experimental investigations into the memory capacity of SSMs have remained relatively cursory. In particular, most prior works have only reported time-averaged benchmark scores, without offering a detailed analysis of the temporal patterns in memorized vs. overlooked information (Gu et al., 2020, 2021, 2022b,a; Gupta et al., 2022). The present study addresses this underexplored question and reveals a primacy effect in SSMs, which stands in stark contrast to their theoretically prescribed memory decay.

2.2. Data-Driven Primacy Effect in Language Models

Several prior studies have documented the primacy effect of artificial neural networks trained on the autoregressive language modeling task. Wang et al. (2023) investigated positional biases in a Transformer-based large language model (LLM) using a prompting-based approach. Specifically, a list of action or event labels was sequentially presented (e.g., "Label 1: change_pin", "Label 2: card_arrival", "Label 3: activate_my_card"). The model was then given a query prompt specifying a target action/event (e.g., "Target Text: I need a new PIN.") along with an instruction statement (e.g., "Which label matches the intent expressed in the Target Text?"). The primacy effect was observed as a greater frequency of the initially presented labels in the model's responses. Comparable findings have been reported across different LLM implementations and benchmark datasets (Eicher and Irgolič, 2024; Guo and Vosoughi, 2024; Janik, 2024; Liu et al., 2024).

Xiao et al. (2024) found that Transformer-based LLMs allocated disproportionate attention to initial tokens, regardless of their informational salience in the text (a phenomenon they termed *attention sinks*). Furthermore, retaining these initial tokens even after they fall outside the predefined input window was found to enhance model performance.

Since the Transformer architecture is inherently position-agnostic—lacking an intrinsic ordering mechanism apart from external positional encodings (Vaswani et al., 2017)—the primacy effects observed in these studies must stem from the statistical properties of the training data or task design. Wang et al. argued that LLMs inherit cognitive biases from human-generated linguistic data. Xiao et al. suggested that the nature of the language modeling task itself encourages prioritization of initial tokens, as they are repeatedly used as inputs for autoregressive predictions, reinforcing attention allocation to them.

In contrast to these prior investigations, the present study examines the emergence of the primacy effect in the SSM while *preventing the inheritance of human-induced bias*. Specifically, the models are trained on a synthetic memorization task designed based on psychological experiments conducted with humans and other animals (Thompson and Herman, 1977; Sands and Wright, 1980; Wright et al., 1985). The following section details the task formulation and the model architecture.

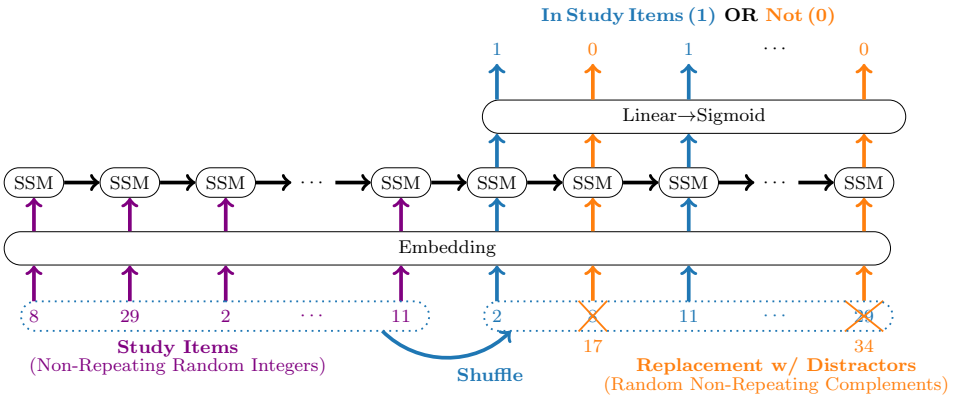

Figure 3: Schematic illustration of the binary memory verification task.

3. Methods

3.1. Task

The memorization patterns of the SSM were assessed using the binary memory verification task (Figure 3; a.k.a. *serial probe recognition* in psychology and ethology; Wickelgren and Norman, 1966; Thompson and Herman, 1977; Sands and Wright, 1980; Wright et al., 1985).[5] In this task, the models were first presented with a sequence of randomly generated, non-repeating integers (hereinafter referred to as *study items*). Subsequently, they received another sequence of integer queries and were trained to determine whether each query token was present (labeled as 1) or absent (labeled as 0) in the study items. To construct these queries, the study items were first shuffled, and then, with a probability of $p = 0.5$, each shuffled token was replaced with a randomly sampled integer from the complement set of the study items (termed *distractors*).[6]

The task hyperparameters were manually adjusted to prevent the models from achieving perfect accuracy. Specifically, the input length was set to $L \in \{64, 128, 256\}$, and the vocabulary size was fixed at $K := 4096$. Each model underwent ten independent training runs with different random seeds. For evaluation, 1024 sets of integers were held out as

5. The most widely adopted task for assessing the primacy effect in human memory is *free recall*, in which participants are presented with a sequence of items and subsequently asked to recall them in an order-agnostic manner (Murdock, 1962; Glanzer and Cunitz, 1966). While this paradigm can be technically formulated as a loss function—under the framework of the optimal transport (Cuturi, 2013)—initial explorations of this study revealed that model performance remained suboptimal under this approach, yielding lower accuracy than in theoretically more demanding tasks requiring order-sensitive reconstruction. Consequently, the present study adopted the more machine learning-friendly task based on binary verification. Remarkably, this task has also been used to assess the memory capacity of non-human animals, which are unable to perform free recall (Thompson and Herman, 1977; Sands and Wright, 1980; Wright et al., 1985).
6. An anonymous reviewer noted that the adopted memorization task introduces a non-uniform distribution of relative distances between study items and queries. Specifically, the model has a lower probability of encountering distances of length $L \pm \alpha$, where L is the number of study items, as α increases from 0 to $L - 1$. Nevertheless, the task preserves *symmetry* between short and long distances, since distances of length $L - \alpha$ and $L + \alpha$ occur with equal frequency. Therefore, the statistical properties of the task do not inherently favor the memorization of earlier study items (i.e., long input-output dependencies).

test data, ensuring that these integer combinations never appeared as study items in the training set, regardless of their order.

To build test sequences, the held-out study items were randomly ordered, and queries were generated by first shuffling and then cyclically shifting them (e.g., $(2, 8, 11, 29) \mapsto \{(2, 8, 11, 29), (8, 11, 29, 2), (11, 29, 2, 8), (29, 2, 8, 11)\}$). This design ensured that each study item was queried in all L possible positions. Finally, either the even- or odd-indexed query positions were replaced with random distractors, resulting in a total of $1024 \times L \times 2$ test sequences per trial.

In Appendix B, the primacy effect is examined in a more advanced task—*associative recall*—which has been established as a useful benchmark for evaluating the performance of language model architectures (Olsson et al., 2022; Fu et al., 2023; Gu and Dao, 2024).

3.2. Models

The models used for the binary memory verification task comprised three layers, as illustrated in Figure 3. In the first layer, the input integers were embedded into 256-dimensional real-valued vectors. These embeddings were shared between study items and query tokens. The resulting sequence of vectors was then processed by the SSM, whose outputs were linearly projected onto binary logits to determine whether each query token was present in the study items.

This study primarily examined the single-layer S4 model as the goldstandard implementation of the SSM (Gu et al., 2022b).[7] The model encoded the channel-wise dynamics of the input embeddings in a complex-valued space, with its outputs subsequently projected back into the real domain by discarding imaginary components. The state and input matrices (A and B in Eq. 3) were initialized to approximate each channel's trajectory using Legendre/Laguerre polynomials of degrees 0–63 (HiPPO-LegS/LagT) or a Fourier basis $\{s_0, c_0, \ldots, s_{31}, c_{31}\}$, where $s_n(t) := \sqrt{2}\sin(2\pi nt)$ and $c_n(t) := \sqrt{2}\cos(2\pi nt)$ (HiPPO-Fout, Fourier Recurrent Unit; Zhang et al., 2018; Gu et al., 2020, 2023). The matrices were discretized by the bilinear method (Tustin, 1947).

For comparison, a single-layer long short-term memory (LSTM) network was also evaluated (Hochreiter and Schmidhuber, 1997). The dimensionality of both hidden and cell states was set to 256.

The models were trained for 300,000 iterations using the Adam optimizer with parameters $(\beta_0, \beta_1) := (0.9, 0.99)$ (Kingma and Ba, 2015). Batch size was set to 512. The learning rate was linearly increased from 0.0 to 0.001 over the first 1,000 iterations (*warmups*) and subsequently decayed according to the cosine annealing schedule (Loshchilov and Hutter, 2017). To prevent gradient explosion, the gradient norm was clipped at 1.0. The Python code for the experiments is available at https://github.com/tkc-morita/primacy-effect.git.

7. Recent studies have shown that the state matrix (A) of S4 can be simplified into a purely diagonal form without compromising performance (S4D; Gu et al., 2022a). By contrast, the original S4 model introduced an additional low-rank component to the diagonal structure (referred to as the Diagonal Plus Low Rank form, or DPLR) to ensure a mathematically well-founded state matrix. Notably, the diagonal variant exhibited a qualitatively similar primacy effect to the DPLR model. Due to the page limitations, results for the diagonal model are omitted from this paper, and all reported findings are based on the DPLR model.

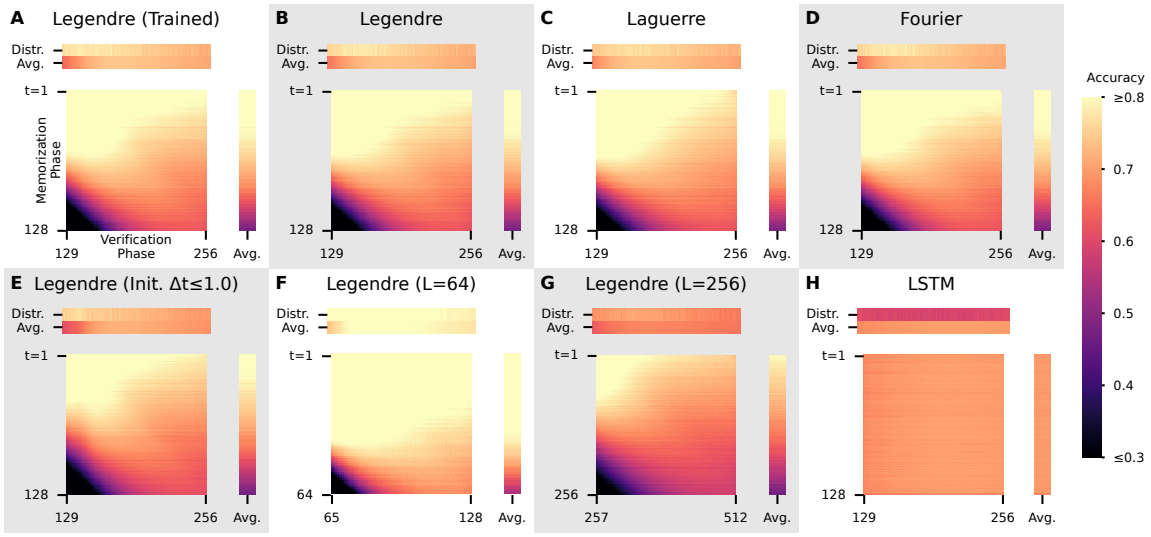

Figure 4: Accuracy of the binary memory verification task. Each cell in the square heatmaps represents the accuracy (or the recall score) for study items that were presented at the time indexed by the corresponding row and queried at the time indexed by the corresponding column. The accuracy for distractor queries is displayed in the top separate row of each panel, alongside the average accuracy across memorization times (rows). Similarly, the rightmost separate column represents the average accuracy across verification times (columns). The bottom-right panel (H) depicts the accuracy distribution for the LSTM, while the other panels (A–G) report results for the SSM (S4) under different parameter configurations. The state and input matrices of the SSM were initialized to approximate the latent dynamics of input sequences using Legendre polynomials, except in panels C and D, where Laguerre and Fourier bases were used, respectively. The state and input matrices were optimized for the task in panel A, whereas they remained fixed at their initial values in all other panels. The discretization step size Δt was initialized in the range $0.001 \leq \Delta t \leq 0.1$, except in panel E, where the upper bound was extended to 1.0 (i.e., $0.001 \leq \Delta t \leq 1.0$). The length of study items was set to $L = 128$, except in panel F ($L = 64$) and panel G ($L = 256$).

4. Results

4.1. Emergence of the Primacy Effect

Figure 4 reports the accuracy of the binary memory verification task across all combinations of memorization and verification times. The brightness of each cell in the square heatmaps indicates the accuracy for study items that were presented at the time indexed by the corresponding row and queried at the time indexed by the corresponding column. That is, they report the proportion of true positives against false negatives (i.e., the *recall* score). Additionally, the top separate row of each panel displays the accuracy for distractor queries (integers not included among the study items), capturing the prevalence of true negatives

over false positives. Just below it, the second row summarizes the average accuracy across memorization times (rows). Similarly, the rightmost separate column represents the average accuracy across all verification times (columns).

The binary memory verification performance of the SSM model was highest for study items presented at the beginning of the sequence, demonstrating a clear primacy effect (Figure 4A–G). The model maintained high accuracy across different query timings (as indicated by the bright colors in the top rows of the heatmaps), provided that the sequence length did not exceed its capacity (see the accuracy decline in Figure 4G, where $L = 256$). In other words, memory for the initial study items exhibited minimal decay over time.

By contrast, the LSTM did not display this primacy effect; its accuracy was uniform across both the memorization and verification phases (Figure 4H).

Interestingly, the SSM's accuracy for the most recently presented study items was lowest when they were queried immediately after their initial presentation in the memorization phase (indicated by the dark colors in the bottom-left region of the heatmaps). This suggests a temporal delay between the encoding of study items and their effective retrieval.

These findings held true regardless of whether the state and input matrices of the SSM (A and B in Eq. 3) were optimized for the task (Figure 4A) or remained fixed at their initial values (Figure 4B–G). Moreover, the results remained consistent across different polynomial bases underlying the state and input matrices, including Laguerre (Figure 4C), Fourier (4D), and Legendre (all other panels).

4.2. Distribution of the Time-Step Sizes

As discussed in §2.1.2, the discretization time-step size Δt plays a critical role in determining the memory capacity of the SSM. Moreover, once the state matrix A is fixed, Δt becomes the sole parameter capable of influencing the *dynamics* of the SSM;[8] all remaining parameters are confined to *feedforward* transforms.

Accordingly, to further investigate its role, the optimization trajectories of Δt were tracked over the course of training. The analysis revealed that as training progressed, a specific range of step sizes ($\Delta t \leq 0.03$) became dominant, compensating for the deallocation of the higher range approximately between 0.03 and 0.2 (Figure 5).

Additionally, the peak value of Δt was found to depend on the number of the study items L; longer study sequences led the model to favor smaller Δt values (compare the leftmost panel with the two rightmost panels).

5. Discussions

The present study demonstrated that the SSM exhibits the primacy effect in memorization. When performing the binary memory verification task, which parallels paradigms used to investigate memory capacity in humans and animals, the model showed the highest accuracy for study items presented at the beginning of the sequence. Moreover, memorized information was not retrievable immediately after the presentation of the study items. These findings are novel and counterintuitive, as they challenge the theoretical formulation of the

8. Freezing Δt resulted in a complete failure of learning.

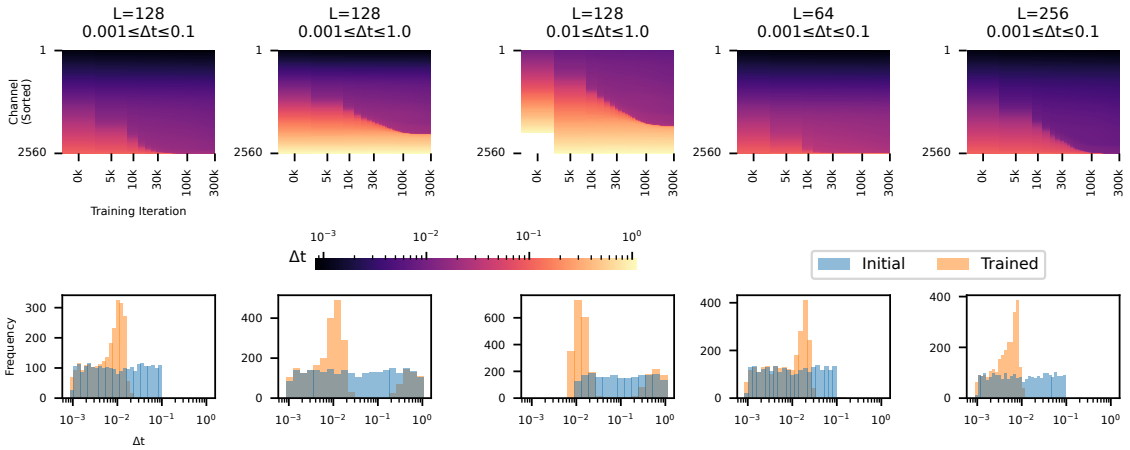

Figure 5: Top row: Optimization trajectories of the discretization step size Δt in the SSM (S4) with frozen Legendre state and input matrices. Each heatmap column represents the distribution of Δt across 256 latent channels $\times$ 10 training runs, sorted in the ascending order. Bottom row: Histograms displaying the initial (blue) and final (orange) values of Δt, aggregated across the 256 latent channels $\times$ 10 training runs (see Figure A.1 in Appendix A for variations among individual runs). The first to third panel columns from the left show the results for three different ranges of log-uniformly random initializations, whereas the fourth and fifth columns tested shorter and longer study items, respectively.

SSM, which assumes an exponentially decaying measure (for Legendre and Laguerre bases; Figure 2; Gu et al., 2020, 2023).

As noted in §2.1.1, the SSM was designed to achieve a longer-lasting memory than classical RNNs (Gu et al., 2020). Prior research on RNNs and SSMs has focused on their ability to preserve input data against temporal decay. However, little attention has been given to how the model handles longer study sequences and larger vocabularies when memory capacity reaches its limit.[9] In particular, the question of whether the models prioritize initial/middle/recent observations has remained unexplored. The present study addressed this question and discovered that the SSM predominantly preserved the initial observations.

The key factor responsible for the primacy effect in the SSM appears to be the time-step size, Δt, as all the other trainable parameters pertain exclusively to feedforward transforms. After training on the memorization task, Δt values concentrated below a specific threshold ($\Delta t \leq 0.03$). As discussed in §2.1.2, smaller Δt values allow the model to retain more distant memories, while larger Δt values enhance the discrimination of adjacent tokens (Gu et al., 2021; Gu and Dao, 2024). The learning results thus align with the *necessary* condition for the primacy effect; however, the question remains open why recent

9. It should be noted that the performance of the SSM can be enhanced by increasing the number of layers and/or latent channels. In this study, the model's capacity was intentionally constrained in order to study its behavior under conditions where perfect accuracy is unattainable.

observations were remembered less accurately despite the exponentially decaying measure underlying the polynomial-approximation theory. Future research may address this issue through comparisons across a wider range of discretization methods—such as the Runge-Kutta method—extending beyond the standard empirical options of bilinear and zero-order hold.

The SSMs analyzed in this study were trained from scratch on a synthetic memorization task that was designed to closely resemble controlled psychological experiments (Wickelgren and Norman, 1966; Thompson and Herman, 1977; Sands and Wright, 1980; Wright et al., 1985). Consequently, the observed primacy effect is attributed to the intrinsic properties of the SSM per se, rather than to biases introduced by data or task design. From this perspective, the the present study stands in contrast to prior investigations of LLMs (Wang et al., 2023; Eicher and Irgolič, 2024; Guo and Vosoughi, 2024; Janik, 2024; Liu et al., 2024; Xiao et al., 2024); LLMs are trained on human-generated linguistic data and therefore likely to inherit the primacy effect as a byproduct of human cognitive biases embedded in the data.

It also remains an open question whether the primacy effect holds in more advanced settings than those examined in this study. Specifically, the scope was restricted to single-layered models, whereas empirical applications almost invariably employ multi-layered architectures. Such extended architectures achieved perfect accuracy on the adopted task—even at the maximal levels of input length and vocabulary size implementable within the available computational resources—thereby failing to incur the memory load necessary for evaluating the primacy effect. For the same reason, the proposed experimental paradigm was also inadequate for testing the SSM-based language model, Mamba (Gu and Dao, 2024; Dao and Gu, 2024). Therefore, Clarifying whether the primacy effect persists in such powerful architectures is an important avenue for future research.

Acknowledgments

This study was supported by JST AIP Accelerated Program (JPMJCR25U6), ACT-X (JPMJAX21AN), and Core Research for Evolutional Science and Technology (JPMJCR22P5); JSPS Grant-in-Aid for Early-Career Scientists (JP21K17805) and for Scientific Research A (JP24H00774), B (JP22H03914), and C (JP24K15087); and Kayamori Foundation of Informational Science Advancement (K35XXVIII620). The author also gratefully acknowledges the support of the Academic Center for Computing and Media Studies, Kyoto University, regarding the use of their supercomputer system.

References

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 10041–10071. PMLR, 21–27 Jul 2024.

Hermann Ebbinghaus. *Memory: A contribution to experimental psychology*. Teachers College Press, 1913. doi: 10.1037/10011-000.

Jonathan E. Eicher and Rafael F. Irgolič. Reducing selection bias in large language models, 2024.

Daniel Y. Fu, Tri Dao, Khaled Kamal Saab, Armin W. Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*. OpenReview.net, 2023.

Murray Glanzer and Anita R. Cunitz. Two storage mechanisms in free recall. *Journal of Verbal Learning and Verbal Behavior*, 5(4):351–360, 1966. ISSN 0022-5371. doi: 10.1016/S0022-5371(66)80044-0.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *Proceedings of the First Conference on Language Modeling*, 2024.

Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. HiPPO: Recurrent memory with optimal polynomial projections. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1474–1487. Curran Associates, Inc., 2020.

Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 572–585. Curran Associates, Inc., 2021.

Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 35971–35983. Curran Associates, Inc., 2022a.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *Proceedings of the Tenth International Conference on Learning Representations (ICLR)*. OpenReview.net, 2022b.

Albert Gu, Isys Johnson, Aman Timalsina, Atri Rudra, and Christopher Ré. How to train your HIPPO: State space models with generalized orthogonal basis projections. In *Proceedings of the Eleventh International Conference on Learning Representations (ICLR)*. OpenReview.net, 2023.

Xiaobo Guo and Soroush Vosoughi. Serial position effects of large language models, 2024.

Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35*, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016. doi: 10.1109/CVPR.2016.90.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735.

Romuald A. Janik. Aspects of human memory and large language models, 2024.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of 3rd International Conference on Learning Representations (ICLR)*, San Diego, California, 2015.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638.

Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*. OpenReview.net, 2017.

Bennet B. Murdock. The serial position effect of free recall. *Journal of Experimental Psychology*, 64(5):482–488, November 1962. ISSN 0022-1015. doi: 10.1037/h0045106.

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. *Transformer Circuits Thread*, 2022.

Stephen F. Sands and Anthony A. Wright. Primate memory: Retention of serial list items by a rhesus monkey. *Science*, 209(4459):938–940, 1980. doi: 10.1126/science.6773143.

Roger K. R. Thompson and Louis M. Herman. Memory for lists of sounds by the bottlenosed dolphin: Convergence of memory processes with humans? *Science*, 195(4277): 501–503, 1977. doi: 10.1126/science.835012.

Arnold Tustin. A method of analysing the behaviour of linear systems in terms of time series. *Journal of the Institution of Electrical Engineers - Part IIA: Automatic Regulators and Servo Mechanisms*, 94:130–142, 1947. doi: 10.1049/ji-2a.1947.0020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017.

Aaron Voelker, Ivana Kajić, and Chris Eliasmith. Legendre memory units: Continuous-time representation in recurrent neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Yiwei Wang, Yujun Cai, Muhao Chen, Yuxuan Liang, and Bryan Hooi. Primacy effect of ChatGPT. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 108–115, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.8.

Wayne A. Wickelgren and Donald A. Norman. Strength models and serial position in short-term recognition memory. *Journal of Mathematical Psychology*, 3(2):316–347, 1966. ISSN 0022-2496. doi: 10.1016/0022-2496(66)90018-6.

Anthony A. Wright, Hector C. Santiago, Stephen F. Sands, Donald F. Kendrick, and Robert G. Cook. Memory processing of serial lists by pigeons, monkeys, and people. *Science*, 229(4710):287–289, 1985. doi: 10.1126/science.9304205.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*. OpenReview.net, 2024.

Jiong Zhang, Yibo Lin, Zhao Song, and Inderjit Dhillon. Learning long term dependencies via Fourier recurrent units. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5815–5823. PMLR, 10–15 Jul 2018.