Generalization Error Analysis for Selective State-Space Models Through the Lens of Attention

Arya Honarpisheh **Mustafa Bozdag** **Octavia Camps** **Mario Sznaier**
Department of Electrical and Computer Engineering
Northeastern University, Boston, MA 02115
{honarpisheh.a, bozdag.m}@northeastern.edu,
{camps, msznaier}@coe.neu.edu

Abstract

State-space models (SSMs) have recently emerged as a compelling alternative to Transformers for sequence modeling tasks. This paper presents a theoretical generalization analysis of selective SSMs, the core architectural component behind the Mamba model. We derive a novel covering number-based generalization bound for selective SSMs, building upon recent theoretical advances in the analysis of Transformer models. Using this result, we analyze how the spectral abscissa of the continuous-time state matrix influences the model's stability during training and its ability to generalize across sequence lengths. We empirically validate our findings on a synthetic majority task, the IMDb sentiment classification benchmark, and the ListOps task, demonstrating how our theoretical insights translate into practical model behavior.

1 Introduction

Foundation models trained on large-scale datasets have become the dominant paradigm in modern machine learning following the introduction of the Transformer architecture [1]. Although they are very effective in sequence processing, the global nature of self-attention imposes limitations: a finite sequence window and quadratic scaling w.r.t. the window length. Recently, a new family of models named state-space models (SSMs) became a popular alternative to address this problem [2, 3]. These models are rooted in the classical state-space representations from control theory introduced by Kalman et al. [4]. Theoretical analysis of SSMs has primarily focused on linear time-invariant (LTI) settings. Rácz et al. [5] derive a length-independent generalization bound for deep stable LTI SSMs using Rademacher contraction techniques, while Liu and Li [6] establishes a length-dependent bound and proposes corresponding initialization and regularization schemes. These results rely on system stability from control theory, ensuring that operator norms remain well-defined and finite. LTI architectures allow for an easy imposition of the stability assumption, making them well-suited for bounded-norm analysis.

The state-of-the-art SSMs that form the core of the Mamba and Mamba-2 architectures are selective and inherently nonlinear [7, 8]. They employ a selective-scan mechanism to capture long-term dependencies, allowing for adaptive and efficient sequence processing. This design, however, makes their analysis distinct from LTI SSMs and more closely aligned with self-attention. Despite their growing empirical success, theoretical understanding of their capabilities remains limited, with the exceptions of Ali et al. [9] and Dao and Gu [8], who establish a connection between Transformers and SSMs, Yu et al. [10] who studies the initialization and training behavior of SSMs, and other recent works that examine their expressive power [11–15].

Alongside expressive power, which characterizes model bias, an equally important question is that of generalization, which captures the variance component of the bias–variance tradeoff, and it is central

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

in theoretical machine learning. Recently, generalization bounds have been developed across various domains, including graph neural networks [16], large language models [17], meta-learning [18], recommender systems [19], representation learning [20], and various neural network architectures [21–23]. Work on the generalization capabilities of attention models often builds on the covering-based framework of Zhang [24] and Bartlett et al. [25], originally developed for linear models and deep networks. Recent adaptations include Edelman et al. [26], which studies inductive bias in Transformers; Trauger and Tewari [27], which establishes length-independent bounds; and Truong [28], which derives rank-dependent bounds.

In this work, we investigate the generalization properties of selective SSMs. Unlike prior work on LTI SSMs that relies on classical tools from control theory and stability assumptions, our approach builds on recent theoretical developments for Transformers and self-attention mechanisms. By unrolling the selective SSM into an attention-like formulation, we enable the use of covering number techniques originally developed for Transformer models. This connection introduces new technical challenges: selective SSMs feature input-conditioned dynamics, input-projected $\boldsymbol{B}$ and $\boldsymbol{C}$ matrices, and discretization from continuous-time systems. We address these by developing a unified covering argument that combines tools from both RNN and attention-based analyses. In particular, we handle the state matrix $\boldsymbol{A}$ through stability and discretization, while treating the input projections $W_{\boldsymbol{B}}$ and $W_{\boldsymbol{C}}$ as linear function classes, analogous to the key-query structure in attention. This results in a two-tiered covering construction that captures the hybrid structure of selective SSMs and is central to our generalization bound, allowing us to study how model characteristics influence generalization and training behavior. We support our theoretical findings with experiments on synthetic and real-world sequence classification tasks. Our main contributions can be summarized as follows:

Theoretically, we derive a novel covering number-based generalization bound for selective SSMs by unrolling their structure and leveraging their connection to attention mechanisms. We also provide a bound for linear attention to rigorously demonstrate the link between the two model classes.

Analytically, we investigate the implications of our bound to understand how generalization depends on the sequence length T and the spectral abscissa $s_{\boldsymbol{A}}$ of the state matrix. In particular, we show how training can fail to stabilize unstable models when T is large, resulting in poor generalization despite the expressivity of the model.

Empirically, we validate our analysis on a synthetic majority task, the IMDb sentiment classification benchmark, and the ListOps task. These experiments demonstrate how model behavior varies with sequence length and stability, highlighting the practical impact of our theoretical insights.

2 Preliminaries

2.1 Notation

We denote the real and imaginary parts of a complex number by $\Re(\cdot)$ and $\Im(\cdot)$. We use $\mathbb{R}, \mathbb{R}^n, \mathbb{R}^{n\times m}$ to denote the set of real numbers, n-dimensional real vectors, and $n \times m$ real-valued matrices, respectively. Lowercase letters like x refer to scalars or vectors, while bold uppercase letters like $\boldsymbol{X}$ denote matrices. The ℓ_p-norm of a vector x is written as $\|x\|_p$, and the p-operator norm of a matrix $\boldsymbol{X}$ is written as $\|\boldsymbol{X}\|_p$. For a matrix $\boldsymbol{X}$ with column vectors $x_1, \ldots, x_n$, the mixed norm $\|\boldsymbol{X}\|_{p,q}$ is defined as $\| \begin{bmatrix}\|x_1\|_p & \cdots & \|x_n\|_p\end{bmatrix} \|_q$. The Kronecker product of matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ is denoted by $\boldsymbol{X} \otimes \boldsymbol{Y}$. Continuous-time state-space matrices are written as $\boldsymbol{A}_c, \boldsymbol{B}_c, \boldsymbol{C}_c$, while their discrete-time counterparts are denoted by $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$. The variable N represents the number of states per channel, T is the sequence length, d is the number of channels, and m is the number of samples. We use (t) to denote continuous time indexing and $[t]$ for discrete time. Lastly, we denote by $s_{\boldsymbol{A}} := \max_i\{\Re(\lambda_i(\boldsymbol{A}))\}$ the *spectral abscissa* of a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, the largest real part among the eigenvalues of $\boldsymbol{A}$. This quantity is commonly used in stability analysis, since a continuous-time system with state matrix $\boldsymbol{A}$ is asymptotically stable in the sense of Lyapunov if and only if $s_{\boldsymbol{A}} < 0$.

2.2 State-Space Models

LTI SSMs are based on the classical single-input single-output (SISO) state-space representation:

$$\begin{aligned} \dot{x}^{(j)}(t) &= \boldsymbol{A}_c^{(j)} x^{(j)}(t) + \boldsymbol{B}_c^{(j)} u_j(t) \\ y_j(t) &= \boldsymbol{C}_c^{(j)} x^{(j)}(t) \end{aligned} \tag{1}$$

where $\boldsymbol{A}_c^{(j)} \in \mathbb{R}^{N\times N}$, $\boldsymbol{B}_c^{(j)} \in \mathbb{R}^{N\times 1}$, $\boldsymbol{C}_c^{(j)} \in \mathbb{R}^{1\times N}$, and $D_c^{(j)} \in \mathbb{R}$ represent the dynamics of a continuous-time LTI system corresponding to the j^{th} channel (feature) $u_j(t)$ of the input signal $u(t)$ using the hidden states $x^{(j)}(t) \in \mathbb{R}^N$. With the notable exception of S5 [29], most SSM implementations use d SISO SSMs as in (1) in a single block, one for each channel respectively.

Selective SSMs, introduced with Mamba [7], make the model parameters and the discretization step size input-dependent to increase expressive power. In particular, for the j^{th} channel, the continuous-time state-space parameters $\boldsymbol{B}_c^{(j)}(t)$, $\boldsymbol{C}_c^{(j)}(t)$, and the step size $\Delta(t)$ are:

$$\begin{aligned} \boldsymbol{B}_c^{(j)}(t) &= \boldsymbol{W}_B\, u(t) \\ \boldsymbol{C}_c^{(j)}(t) &= u(t)^\top \boldsymbol{W}_C^\top \end{aligned} \qquad \Delta(t) = \tau_\Delta\big(p + q^\top u(t)\big) \tag{2}$$

Here, $\boldsymbol{W_B}, \boldsymbol{W_C} \in \mathbb{R}^{N\times d}, q \in \mathbb{R}^d$ and $p \in \mathbb{R}$ are learnable weights and $\tau_\Delta(x) = \ln(1+e^x)$ is the softplus function. The input and output matrices $\boldsymbol{B}_c^{(j)}$ and $\boldsymbol{C}_c^{(j)}$ are linear projections of the input $u(t)$. Thus, the state-space model for the j^{th} channel is:

$$\begin{aligned} \dot{x}^{(j)}(t) &= \boldsymbol{A}_c^{(j)} x^{(j)}(t) + \boldsymbol{W_B} u(t) u_j(t) \\ y^{(j)} &= u^\top \boldsymbol{W_C}^\top x^{(j)}(t). \end{aligned} \tag{3}$$

To derive a state-space model including all states and inputs, we stack the states in (3) to get one single state vector $x = \big[(x^{(1)})^\top \ \dots \ (x^{(d)})^\top\big]^\top \in \mathbb{R}^{Nd}$ that satisfies the following state-space model:

$$\begin{aligned} \dot{x}(t) &= \boldsymbol{A}_c x(t) + \boldsymbol{B}_c u(t) \\ y(t) &= \boldsymbol{C}_c x(t) \end{aligned} \tag{4}$$

in which

$$\boldsymbol{A}_c = \begin{bmatrix} \boldsymbol{A}^{(1)} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}^{(2)} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{A}^{(d)} \end{bmatrix}, \quad \begin{aligned} \boldsymbol{B}_c(t) &= \mathbf{I}_d \otimes \boldsymbol{W_B} u(t) \\ \boldsymbol{C}_c(t) &= \mathbf{I}_d \otimes u(t)^\top \boldsymbol{W_C}^\top \end{aligned} \tag{5}$$

For the discretization step, we follow the official implementation of Mamba [7]:

$$\boldsymbol{A}[t] = \exp\big(\Delta(t)\boldsymbol{A}_c\big), \quad \begin{aligned} \boldsymbol{B}[t] &= \Delta(t)\boldsymbol{B}_c \\ \boldsymbol{C}[t] &= \boldsymbol{C}_c \end{aligned} \tag{6}$$

which uses a zero-order hold (ZOH) for the matrix $\boldsymbol{A}$, and a simplified Euler discretization for the matrix $\boldsymbol{B}$. The matrix $\boldsymbol{C}$ remains the same as $\boldsymbol{C}_c$ since it represents a static relationship between the output y and state x. Utilizing this discretization procedure and the selective SSM architecture in (4) results in the following discrete-time state-space model:

$$\begin{aligned} x[t+1] &= e^{\Delta[t]\boldsymbol{A}_c} x[t] + \Delta[t]\big(\mathbf{I}_d \otimes \boldsymbol{W_B} u[t]\big) u[t] \\ y[t] &= \big(\mathbf{I}_d \otimes u[t]^\top \boldsymbol{W_C}^\top\big) x[t] \\ \Delta[t] &= \tau_\Delta(p + q^\top u[t]). \end{aligned} \tag{7}$$

Assuming $x(0) = 0$, we can unroll this recursive relation:

$$y[t'] = \big(\mathbf{I}_d \otimes u[t']^\top \boldsymbol{W_C}^\top\big) \sum_{t=0}^{t'-1} \Big(\boldsymbol{A}^t \Delta[t'-1-t]\big(\mathbf{I}_d \otimes \boldsymbol{W_B} u[t'-1-t]\big) u[t'-1-t]\Big) \tag{8}$$

in which $\boldsymbol{A}^t$ is a shorthand notation for

$$\boldsymbol{A}^0 = \mathbf{I}, \quad \boldsymbol{A}^t = e^{(\Delta[t'-1]+\cdots+\Delta[t'-t])\boldsymbol{A}_c} \text{ for } t > 0. \tag{9}$$

For classification tasks requiring a scalar output, an additional parameter $w \in \mathbb{R}^d$ is introduced. This parameter corresponds to a linear layer applied to the last time step of the output sequence to obtain a label as $z = w^\top y[T]$, which captures all past information. The space of all selective SSMs $\mathcal{F}_{\text{SSM}}$ as defined in (7) is parametrized by

$$\Theta_{\text{SSM}} = \{\boldsymbol{A}_c, \boldsymbol{W_B}, \boldsymbol{W_C}, p, q, w\}. \tag{10}$$

3 Generalization Bounds for Selective SSMs

To quantify generalization, we study the gap between the empirical training loss and the true expected loss over unseen data. Following standard results from statistical learning theory, we upper bound this generalization gap using the Rademacher complexity of the function class. Rademacher complexity of a function class $\mathcal{F}$ for a given dataset S, denoted as $\mathrm{Rad}(\mathcal{F}, S)$, measures the average ability of a function class to fit random noise and serves as a data-dependent notion of capacity. For completeness, we include the formal definition of Rademacher complexity and the precise statement of the theorem that bounds the generalization gap in terms of it in Appendix E.

Bounding the Rademacher complexity of complex function classes, such as foundation models, is challenging. A common approach is to break down the model into smaller components and analyze each of them separately by employing covering numbers. Covering numbers quantify the size of each component's function class by determining how many smaller subsets, or "balls", are needed to cover it. Formally, the covering number is defined as follows.

Definition 3.1 (**Covering number**). *Let $\mathcal{M}$ be a metric space with metric μ. A subset $\mathcal{H} \subset \mathcal{M}$ is an ϵ-cover for $\mathcal{M}$ if for every $h \in \mathcal{M}$, there exists $\hat{h} \in \mathcal{H}$ such that $\mu(h, \hat{h}) \leq \epsilon$. The covering number $\mathcal{N}(\mathcal{M}, \epsilon, \mu)$ is the lowest cardinality of an ϵ-cover of $\mathcal{M}$.*

Remark 3.1 (**Types of covering numbers**). In our analysis, two distinct types of covering numbers are employed. The metric space $\mathcal{M}$ in Definition 3.1 can be chosen as a collection of matrices equipped with the metric induced by a matrix norm. We deploy this definition to construct a cover for the parameters $\boldsymbol{A}_c$ and p, similar to the existing covering techniques developed for RNNs. On the other hand, let $\mathcal{F} = \{f : \mathcal{U} \to \mathcal{Z}\}$ denote a function class, where $\mathcal{Z}$ is equipped with a norm $\|\cdot\|$, and let $S = \{u_{(i)}\}_{i=1}^m$ be a dataset. By equipping $\mathcal{F}$ with the following metric

$$\mu_{p,\|\cdot\|}(f, \hat{f}) = \left(\frac{1}{m} \sum_{i=1}^{m} \left\| f(u_{(i)}) - \hat{f}(u_{(i)}) \right\|^p \right)^{1/p}, \tag{11}$$

we obtain a *data-dependent* covering number for a function class, to construct covers for $\boldsymbol{W_B}, \boldsymbol{W_C}, q$, and w. This is parallel to the cover construction for the query, key, and value weight matrices in self-attention. For convenience, we denote the covering number of a function class $\mathcal{N}(\mathcal{F}, \epsilon, \mu_{p,\|\cdot\|})$ by $\mathcal{N}_p(\mathcal{F}_{|S}, \epsilon, \|\cdot\|)$.

The covering numbers outlined in Remark (3.1) can be utilized to bound the Rademacher complexity of a selective SSM parametrized as in (10) via Dudley's integral theorem, stated below.

Theorem 3.2 (Bartlett et al. [25], Lemma A.5). *Given a real-valued function class $\mathcal{F} = \{f : \mathcal{U} \to \mathbb{R}\}$ such that $\forall u \in \mathcal{U},\ |f(u)| \leq \mathfrak{b}$ and a set of vectors $S = \{u_{(i)}\}_{i=1}^m$, we have*

$$\mathrm{Rad}(\mathcal{F}, S) \leq \inf_{\alpha > 0} \left(4\alpha + 12 \int_{\alpha}^{\mathfrak{b}} \sqrt{\frac{\ln \mathcal{N}_2(\mathcal{F}_{|S}, \epsilon, \|\cdot\|_2)}{m}}\, d\epsilon \right). \tag{12}$$

[1]

3.1 Main Result

In this section, we present an upper bound on the generalization error for selective SSMs. Before that, we outline and justify the assumptions necessary for deriving the bound. These assumptions are needed mostly to construct suitable covers through the lemmas presented in Appendix D.

Assumption 3.1 (Input). The input sequence $\|u[t]\|_2 \leq \mathfrak{B}_u$ for all t.

Assumption 3.2 (Parameters). The parameters obey $\|\boldsymbol{W}_B\|_2 \leq \mathfrak{B}_{\boldsymbol{B}}$, $\|\boldsymbol{W}_B\|_{1,1} \leq \mathfrak{M}_{\boldsymbol{B}}$, $\|\boldsymbol{W}_C\|_2 \leq \mathfrak{B}_{\boldsymbol{C}}$, $\|\boldsymbol{W}_C\|_{1,1} \leq \mathfrak{M}_{\boldsymbol{C}}$, $\|w\|_2 \leq \mathfrak{B}_w$, $\|w\|_1 \leq \mathfrak{M}_w$, $\|q\|_2 \leq \mathfrak{B}_q$, and $\|q\|_1 \leq \mathfrak{M}_q$.

Assumption 3.3 (Loss function). The loss function $l(\cdot)$ is bounded by $\mathfrak{c}_l$ and Lipschitz continuous with constant $\mathfrak{l}_l$.

Assumption 3.4 (State Matrix $\boldsymbol{A}_c$). The continuous-time state matrix $\boldsymbol{A}_c$ satisfies $\|\boldsymbol{A}_c\|_2 \leq \mathfrak{B}_{\boldsymbol{A}}$ and $\|\boldsymbol{A}_c\|_{2,1} \leq \mathfrak{M}_{\boldsymbol{A}}$.

[1] This is a modified version of Dudley's integral theorem [30]. The proof presented in [25] ignored the normalization by m in (11) and takes $\mathfrak{b} = 1$ in (12).

The bounds on the norms $\|\cdot\|_2$, $\|\cdot\|_1$, and $\|\cdot\|_{1,1}$ in Assumptions 3.1 and 3.2 are standard and chosen to enable the use of Lemma D.3, adapted from [27] originally developed for Transformers. This result provides a covering number bound for linear function classes with bounded $\|\cdot\|_{1,1}$ norm, without introducing sample size or sequence length dependencies. While alternative norms can be used, they typically lead to looser or length-dependent bounds. The Lipschitz continuity of the loss function in Assumption 3.3 ensures that small changes in the input lead to controlled changes in the loss, which is essential for bounding the model's generalization error in Lemma D.4. Assumption 3.4 employs different norms than 3.2 ($\|\cdot\|_{2,1}$ rather than $\|\cdot\|_{1,1}$) because we cover $\boldsymbol{A}_c$ with a distinct strategy (Lemma D.2) from the other parameters. With these in place, we now state the following theorem, whose proof is provided in Appendix E.

Theorem 3.3. *Let $S = \{u_{(i)}, z_{(i)}\}_{i=1}^m$ be the training set and $\mathcal{F}_{SSM}$ be all selective SSM blocks described in (7). If Assumptions 3.1–3.4 are satisfied, then with probability more than $1-\delta$ the following bound for any $h \in \mathcal{F}_{SSM}$ holds:*

$$\left|\mathbb{E}_{u,z}(l(h(u),z)) - \frac{1}{m}\sum_{i=1}^{m} l\left(h(u_{(i)}), z_{(i)}\right)\right| \leq \frac{12\mathfrak{l}_l \mathcal{C}_{\mathcal{F}_{SSM}}}{\sqrt{m}}\left(1+\ln\left(\frac{\mathfrak{c}_l\sqrt{m}}{3\mathcal{C}_{\mathcal{F}_{SSM}}}\right)\right) + 3\mathfrak{c}_l\sqrt{\frac{\ln\left(\frac{2}{\delta}\right)}{2m}} \tag{13}$$

in which

$$\mathcal{C}_{\mathcal{F}_{SSM}} = \tilde{\mathcal{O}}\left(\mathfrak{M}_\Delta \mathfrak{B}_w \mathfrak{B}_u^3 \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_{\boldsymbol{A}} S_2 (\mathfrak{M}_\Delta^{2/3} N^{1/3} d^{1/3} + \mathfrak{B}_q^{2/3}\mathfrak{B}_u^{2/3})^{3/2}\right) \tag{14}$$

and

$$S_2 = \frac{\rho_{\boldsymbol{A}}(1-\rho_{\boldsymbol{A}}^T)}{(1-\rho_{\boldsymbol{A}})^2} - \frac{T\rho_{\boldsymbol{A}}^T}{\rho_{\boldsymbol{A}}-1}, \quad \rho_{\boldsymbol{A}} = \left(1+e^{p-\mathfrak{B}_q\mathfrak{B}_u}\right)^{s_{\boldsymbol{A}}+\eta}, \quad \mathfrak{M}_\Delta = \ln(1+e^{p+\mathfrak{B}_q\mathfrak{B}_u}), \tag{15}$$

where $\eta > 0$ is any arbitrarily small positive number chosen in a way that $s_{\boldsymbol{A}} + \eta \neq 0$. The notation $\tilde{\mathcal{O}}(\cdot)$ ignores the logarithmic dependencies on N and d, but not T. The terms $\mathfrak{M}_{(\cdot)}$ does not appear in the capacity expression $\mathcal{C}_{\mathcal{F}_{\text{SSM}}}$, with the exception of $\mathfrak{M}_\Delta$. This is the result of an assumption $\mathfrak{M}_{(\cdot)} = \mathfrak{B}_{(\cdot)}$, made for the ease of presentation in the proof. To ensure clarity in the derivation, these terms are handled separately throughout the proof, and the assumption is incorporated only at the final stage.

Proof Sketch of Theorem 3.3. The class of all Selective SSMs, denoted by $\mathcal{F}_{\text{SSM}}$, is parameterized by $\Theta_{\text{SSM}} = \{\boldsymbol{A}_c, \boldsymbol{W_B}, \boldsymbol{W_C}, p, q, w\}$. We bound the generalization gap $|\mathbb{E}(l(h(u),z)) - \frac{1}{m}\sum_{i=1}^m l(h(u^{(i)}), z^{(i)})|$ by controlling the Rademacher complexity of $\mathcal{F}_{\text{SSM}}$.

The first challenge is to bound the distance between $\boldsymbol{A}^t$ and its cover $\hat{\boldsymbol{A}}^t$. Unlike LTI SSMs and RNNs, where the fixed number $\|\boldsymbol{A}\|_2^t$ bounds $\|\boldsymbol{A}^t\|_2$, here $\|\boldsymbol{A}^t\|_2$, as defined by the shorthand in (9), has an input-dependent structure and must be controlled via the learnable parameters $\{p, q, \boldsymbol{A}_c\}$. In Lemma E.3 we construct the appropriate $\rho_{\boldsymbol{A}}$ (as defined in (15)) using Gelfand's formula (Corollary 5.6.14 in [31]) and bounds on $p, q, \boldsymbol{A}_c$ to show $\|\boldsymbol{A}^t\|_2 \leq \rho_{\boldsymbol{A}}^t$, revealing the role of the spectral abscissa of $\boldsymbol{A}_c$. Gelfand's formula characterizes the growth of $\|\boldsymbol{A}^t\|_2$ in terms of the spectral radius of $\boldsymbol{A}$, which is why the small positive number $\eta > 0$ appears in (15). Then Lemma E.5 establishes an upper bound on $\|\boldsymbol{A}^t - \hat{\boldsymbol{A}}^t\|_2$ through a telescoping-sum argument. Finally, Lemma E.6, followed by Lemma E.7, shows how the cover radius for the whole model is decomposed as the sum of the cover radii of each parameter in Θ_{SSM}. Additionally, the structure of the input-dependent time step $\Delta[t]$ should be taken into account, as it affects both $\boldsymbol{A}[t]$ and $\boldsymbol{B}[t]$. The bounded norm assumptions on the input and q, together with the presence of the bias term p in $\Delta[t]$, ensure the existence of a uniform lower bound on the step size. As shown in Lemma E.3, this bound is essential for the application of Gelfand's formula to control the growth of $\|\boldsymbol{A}^t\|_2$.

Next, individual covers are constructed: $\boldsymbol{A}_c$ is covered in Lemma D.2 as an element of the matrix space equipped with its matrix norm. Recognizing the attention mechanism in (8), Lemma D.3 covers the parameters $\{\boldsymbol{W}_B, \boldsymbol{W}_C, q, w\}$ by treating them as linear function classes. These individual covers are combined via a Cartesian product to cover Θ_{SSM}, and the cover radii are chosen optimally according to Lemma C.9 to yield a global ϵ-cover of $\mathcal{F}_{\text{SSM}}$. Finally, Lemma D.4, a standard corollary of the Dudley integral bound on Rademacher complexity, yields the claimed bound. □

Remark 3.2 (Lens of Attention). A novel component of the proof of Theorem 3.3 lies in recognizing the attention mechanism in (8), which guides us to cover the parameter set $\{\boldsymbol{W}_B, \boldsymbol{W}_C, q, w\}$ as a function class. In particular, the line of work by [24, 26, 27] on generalization in Transformers presents numerous covering-number bounds for such parameters. We utilize these results to establish our theorem, obtaining a length-independent generalization bound under the condition $s_{\boldsymbol{A}} < 0$. This result is attainable only by revealing the underlying attention mechanism. To show this connection explicitly, we make the following assumptions to single out the attention in selective SSMs:

Assumption 3.5. Set $q = 0$ and $p = e$, yielding a constant step size $\Delta[t] = 1$.

Assumption 3.6. Assume $\boldsymbol{A}_c = \mathbf{0}$, so that $\boldsymbol{A}[t] = \mathbf{I}$ for all t.

Under Assumptions 3.5 and 3.6, the selective SSM reduces to linear attention with a causal mask [8]. In particular, the output at time T admits the representation

$$z = w^\top \sum_{t=0}^{T-1} \underbrace{\left(\mathbf{I}_d \otimes u[T]^\top \boldsymbol{W}_C^\top\right)}_{\text{Query}} \underbrace{\left(\mathbf{I}_d \otimes \boldsymbol{W}_B\, u[T-1-t]\right)}_{\text{Key}} \underbrace{u[T-1-t]}_{\text{Value}}. \tag{16}$$

By fixing $\boldsymbol{A}[t] = \mathbf{I}$ and $\Delta[t] = 1$, we obtain a significantly simpler derivation of the generalization error bound for linear attention given below, with its proof presented in Appendix F.

Proposition 3.4. *Let $S = \{u_{(i)}, z_{(i)}\}_{i=1}^m$, and suppose Assumptions 3.1–3.4 hold with the simplifications made in Assumptions 3.5 and 3.6. Then, for the class of linear attentions $\mathcal{F}_{LA}$ parametrized by $\{\boldsymbol{W}_C, \boldsymbol{W}_B, w\}$, with probability at least $1-\delta$, the generalization bound in (13) holds, where $\mathcal{C}_{\mathcal{F}_{\text{SSM}}}$ is replaced by*

$$\mathcal{C}_{\mathcal{F}_{LA}} = \tilde{\mathcal{O}}\left(T \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3\right). \tag{17}$$

4 Analysis

In this section, we draw insights from Theorem 3.3: we interpret the dependency of the bound on its parameters and compare it to similar bounds derived for other architectures. Table 1 summarizes the dependency of the generalization error bound on the sequence length T, the input dimension d, and the input magnitude $\mathfrak{B}_u$ for different architectures.

Length Independence. As shown in Theorem 3.3 and Table 1, when the spectral abscissa satisfies $s_{\boldsymbol{A}} < 0$, the generalization bound is length-independent, whereas if $s_{\boldsymbol{A}} > 0$, the bound grows exponentially in T. This aligns with prior results for RNNs [32, 33] and LTI SSMs [5, 34], which similarly emphasize the importance of the Lipschitz constant of the activation function and norm of the state matrix. In our setting, the Mamba construction (6) ensures that the stability properties of the continuous-time matrix $\boldsymbol{A}_c$ carry over to the discrete-time matrix $\boldsymbol{A}[t]$, directly controlling the generalization gap.

Attention. We now compare the bounds we derived in Theorem (3.3) for selective SSMs and Proposition (3.4) for linear attention. The gap between their capacity terms, $\mathcal{C}_{\mathcal{F}_{\text{SSM}}}$ and $\mathcal{C}_{\mathcal{F}_{\text{LA}}}$, arises from simplifying assumptions used to reduce a selective SSM to a linear attention structure. First, Assumption 3.5 removes dependence on the input-conditioned discretization $\Delta[t]$, eliminating the $\mathfrak{M}_\Delta$ term and its associated input-norm $\mathfrak{B}_u$ scaling, resulting in a $\mathfrak{B}_u^3$ dependence—reflecting the three linear projections of attention (query, key, value). In contrast, selective SSMs incur an extra $\mathfrak{B}_u$ factor embedded in $\mathfrak{M}_\Delta$ due to dynamic input influence. Second, Assumption 3.6 further simplifies the bound by removing the need to cover the state matrix $\boldsymbol{A}$, reducing the S_2 term to a linear dependence on T due to projection accumulation. Lastly, comparing linear to softmax attention, the key difference lies in normalization. Softmax reweighs the sequence, removing any explicit T-dependence in the bound and effectively decoupling capacity from sequence length.

RNNs. Consider the vanilla RNN model

$$x[t] = \sigma_x\left(\boldsymbol{A}\, x[t-1] + \boldsymbol{B}\, u[t]\right), \quad y[t] = \sigma_y\left(\boldsymbol{C}\, x[t]\right), \tag{18}$$

where σ_x is $\mathfrak{l}_x$-Lipschitz and bounded. When $\mathfrak{l}_x \|\boldsymbol{A}\|_2 < 1$, the bound is length-independent, whereas for $\mathfrak{l}_x \|\boldsymbol{A}\|_2 \geq 1$ it exhibits linear dependence on T. The key mechanism RNNs avoid exponential dependence is the bounded activation [36]. Unfortunately, incorporating a bounded activation into selective SSMs degrades their training efficiency, the very advantage that motivated their design.

Table 1: **Generalization bound scaling for different sequence-to-sequence models.** Dependencies are shown with respect to sequence length T, hidden dimension d, and input magnitude $\mathfrak{B}_u$ (logarithmic terms in d and $\mathfrak{B}_u$ omitted). † The term $\rho_{\boldsymbol{A}}$ depends on the spectral abscissa $s_{\boldsymbol{A}}$, as defined in (15). When $s_{\boldsymbol{A}} < 0$, we can choose $\eta > 0$ to be arbitrarily small so that $\rho_{\boldsymbol{A}} < 1$, yielding a length-independent bound. ‡ $\mathfrak{l}_x$ is the Lipschitz constant of the activation function used in the RNN as defined in (18). § The bounds for LTI SSMs are derived under different assumptions as explained in Remark 4.1.

Model	Specification	T	d	$\mathfrak{B}_u$
Selective SSM (Theorem 3.3)	$s_{\boldsymbol{A}} < 0$	1	$d^{1/2}$	$\mathfrak{B}_u^4$
	$s_{\boldsymbol{A}} \geq 0$	$T\rho_{\boldsymbol{A}}^T$ †	$d^{1/2}$	$\mathfrak{B}_u^4$
Linear Attention (Proposition 3.4)	N.A.	T	1	$\mathfrak{B}_u^3$
Softmax Attention [27]	N.A.	1	1	$\mathfrak{B}_u^3$
Vanilla RNN [32]	$\mathfrak{l}_x\|\boldsymbol{A}\|_2 < 1$ ‡	1	d	$\mathfrak{B}_u$
	$\mathfrak{l}_x\|\boldsymbol{A}\|_2 = 1$	T	d	$\mathfrak{B}_u$
	$\mathfrak{l}_x\|\boldsymbol{A}\|_2 > 1$	T	$d^{3/2}$	$\mathfrak{B}_u$
Discrete-time LTI SSMs [5] §	$\max_i \lvert\lambda_i(\boldsymbol{A})\rvert < 1$	1	–	Bounded $\sum_{k=1}^{\infty} \|u[k]\|_2^2$
Continuous-time LTI SSMs [35] §	$s_{\boldsymbol{A}} < 0$	$\ln^{3/2}(T)$	–	Holder continuous

Another distinction is that RNN bounds hinge on contractivity—the induced norm of the matrix being less than one or related to the singular values—while our bound depends on the spectral properties of $\boldsymbol{A}_c$. This difference stems from our analysis, which begins with a continuous-time state-space model and discretizes via the matrix exponential, an operation governed by eigenvalues rather than singular values. We believe that the distinctive parameter for RNNs ($\mathfrak{l}_x\|\boldsymbol{A}\|_2$) can be improved in a similar way to our approach, where we took advantage of Gelfand's formula. This allows replacing $\|\boldsymbol{A}\|_2$ with $\max_i |\lambda_i(\boldsymbol{A})|$, which is an improvement since the spectral radius of a matrix is a lower bound on any induced matrix norm.

Remark 4.1 (LTI SSMs). One of the recent generalization bounds for LTI SSMs is the length-independent result of Rácz et al. [5], which applies exclusively to stable discrete-time LTI systems. Their bound relies on the ℓ_1-norm of the system's impulse response and the $\mathcal{H}_2$-norm of its transfer function, both of which are finite for only strictly stable systems. Extending this result to selective SSMs is nontrivial, as there is no direct analogue of these norms for general nonlinear systems. Moreover, they assume that the input satisfies $\sum_{k=1}^{\infty} \|u[k]\|_2^2 < \infty$, which is generally not satisfied in deep learning applications unlike our Assumption 3.1. Liu and Li [6] derive another generalization bound for LTI SSMs, but they aim to consider the temporal dependencies in the input signal via its mean and variance, leading to a Hölder continuity condition. Their bound, under a stability assumption, scales logarithmically with T and is derived for continuous-time SSMs. Thus, in practice, it only applies when the discretized model remains close to its continuous-time counterpart. In contrast, we directly consider a continuous-time SSM with an input-dependent time scale, and derive a generalization bound for the resulting discrete-time model—an assumption more aligned with how such models are used in practice. In summary, while existing techniques for SSMs leverage system-theoretic properties unique to LTI systems, the nonlinearity of selective SSMs requires deriving bounds that are better suited to modern deep learning architectures.

Overall, our results complete the picture of generalization for deep sequence models. Strongest generalization occurs in selective SSMs with $s_{\boldsymbol{A}} < 0$, softmax-attention, and RNNs with $\mathfrak{l}_x\|\boldsymbol{A}\|_2 < 1$, all benefiting from implicit normalization or stabilization. In contrast, selective SSMs with $s_{\boldsymbol{A}} > 0$ generalize poorly. We revisit this in the next section, showing that no sub-exponential bound can be derived using Rademacher complexity in this regime. Later on in Section 5, we observe that the training error grows rapidly unless $s_{\boldsymbol{A}}$ is driven negative, as seen in Figure 1.

4.1 A Lower Bound on the Rademacher Complexity

In this section, we present a theorem that establishes a lower bound on the Rademacher complexity for the case where the spectral abscissa satisfies $s_{\boldsymbol{A}} \geq 0$.

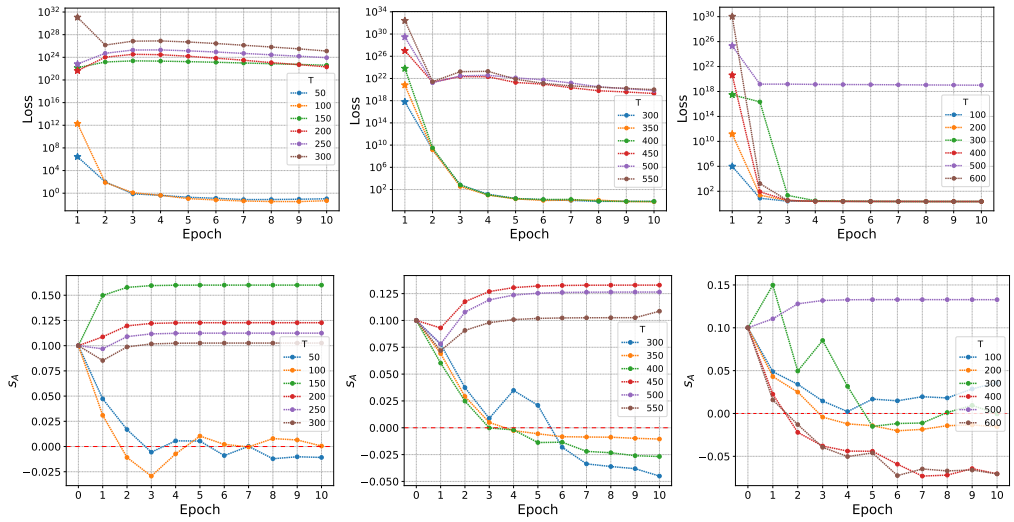

Figure 1: **Experiment 1. Top:** Training loss vs epochs for **Left:** Majority, **Middle:** IMDb, **Right:** ListOps. **Bottom:** Evolution of $s_{\boldsymbol{A}}$ vs epochs for the same datasets. All runs use an *unstable initialization* with $s_{\boldsymbol{A}} = 0.1$. Whenever training successfully reduces the loss, the spectral abscissa $s_{\boldsymbol{A}}$ is driven toward zero, indicating that the system becomes stable. In cases where $s_{\boldsymbol{A}}$ does not decrease toward zero, training is not successful.

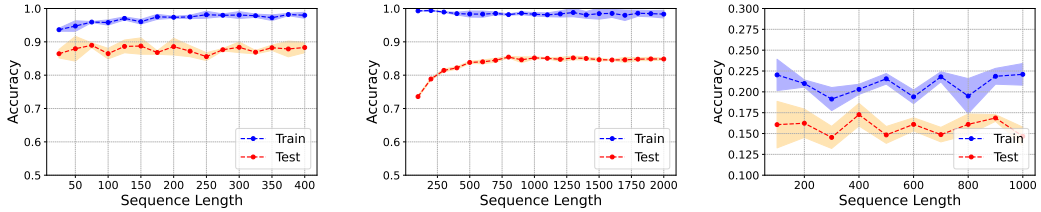

Figure 2: **Experiment 2. Left:** Majority, **Middle:** IMDb, **Right:** ListOps. Train and test accuracy versus sequence length T for models initialized with $s_{\boldsymbol{A}} = 0$. The results demonstrate length-independent generalization. Each experiment is repeated five times with different random seeds; the dashed line denotes the mean accuracy across runs, and the shaded region represents $\pm$ one standard deviation.

Theorem 4.1. *Let $s_{\boldsymbol{A}} > 0$ be a fixed spectral abscissa and $\mathcal{S} = \{u_{(i)}\}_{i=1}^{m} \subset [-1, 1]^T$. If $|w| \leq \mathfrak{B}_w$, then*

$$\mathrm{Rad}_{\mathcal{S}}(\mathcal{F}_{SSM}) \geq \mathfrak{B}_w \frac{(1 + s_{\boldsymbol{A}})^T - 1}{s_{\boldsymbol{A}}} \sqrt{\frac{2}{\pi m}}. \tag{19}$$

Moreover, if $s_{\boldsymbol{A}} = 0$, then

$$\mathrm{Rad}_{\mathcal{S}}(\mathcal{F}_{SSM}) \geq \mathfrak{B}_w T \sqrt{\frac{2}{\pi m}}. \tag{20}$$

The proof of the theorem is provided in Appendix G. It is based on the construction of a restricted subclass of selective SSMs in which the step size is fixed and the output grows unboundedly as a function of a single parameter w. In this simplified setting, the Rademacher complexity can be computed explicitly, yielding a lower bound on the Rademacher complexity of the broader class $\mathcal{F}_{\mathrm{SSM}}$. Unfortunately, these lower bounds are looser than the corresponding upper bounds by a factor of $\mathcal{O}(T)$, suggesting that there is room for improvement in at least one of the bounds. Nevertheless, they demonstrate an important fact: **the dependence on T in the Rademacher complexity cannot be eliminated** when $s_{\boldsymbol{A}} \geq 0$. Specifically, it must grow at least exponentially when $s_{\boldsymbol{A}} > 0$.

5 Experiments

To validate our theoretical findings, we conduct two sets of experiments on three datasets [2]. The first experiment examines the effect of a positive spectral abscissa. It demonstrates that when the system matrix is initialized with positive $s_{\boldsymbol{A}}$, training may fail, especially for longer sequences, rendering the generalization gap ill-defined. Since this gap is defined as the difference between train and test error, if one of these two terms is not well-defined, it would be meaningless to evaluate their difference. In the second experiment, we initialize the state matrix with $s_{\boldsymbol{A}} = 0$ and study how the generalization gap evolves with increasing sequence length. The model architecture used in the experiments consists of an embedding layer, followed by a single selective SSM block. The model is trained using cross-entropy loss. To stabilize training, we employ a regularization function from Keller [37]. Below, we describe the tasks with their corresponding datasets, followed by our analysis of each experiment.

Majority: The first task employs a synthetic "majority" dataset, where each input sequence consists of binary symbols $\{0, 1\}$, embedded into a d-dimensional vector. The objective is to predict whether the sequence contains more ones than zeros. The output depends uniformly on all input positions: no single position has disproportionate influence, and the prediction is not determined by a small subset of the input sequence. This makes the task well-suited for measuring trends in generalization gap w.r.t. the sequence length T. During training, noise is introduced by randomly flipping a small percentage of the inputs after labeling, adding a layer of difficulty. The noise limits the model's accuracy around 95%, preventing it from overfitting despite the simplicity of the task.

IMDb: The second task is binary sentiment classification using the IMDb large movie review dataset [38] containing 50K reviews. Each review is labeled as positive or negative based on its sentiment. This task poses a real-world challenge due to its high variability in sequence lengths and the need for contextual understanding. To control sequence length during training and evaluation, we pad/truncate sequences to a fixed T. For shorter sequences ($T \leq 300$), sentiment indicators are often clear early, aiding prediction, while longer sequences often require retaining more context for accuracy. Thus, truncating them leads to a substantial decrease in model performance, as seen in the test loss in Figure 2. The average review length in the dataset is around $T = 300$. Therefore, the test loss and generalization gap both stabilize after $T = 300$, indicating that enough information is preserved for the model to generalize effectively. This is explained thoroughly in Appendix B.2.

ListOps: The third task uses the ListOps dataset [39], a synthetic benchmark that evaluates a model's ability to reason over hierarchical sequences. Each input is a bracketed expression with nested operations, for example `[MIN 5 1 [MAX 2 9] 0]`, which evaluates to a single-digit integer. The challenge lies in the fact that the correct output depends on the entire nested structure rather than local context. We use the version from Tay et al. [40] to align with standard long-sequence benchmarks.

Experiment 1 (Stability Under Training): This experiment investigates the behavior of the spectral abscissa $s_{\boldsymbol{A}}$ during training. To better understand the role of stability in training selective SSMs, we deliberately initialize all models in an *unstable* regime with $s_{\boldsymbol{A}} = 0.1$. The key observation here is that successful training is accompanied by $s_{\boldsymbol{A}}$ being driven toward zero. We observe that as the sequence length T increases, the initial loss grows exponentially, making it increasingly difficult for the model to escape instability. This ultimately causes training to fail for longer sequences. Since successful training consistently corresponds to $s_{\boldsymbol{A}}$ approaching zero, we consider the regime with $s_{\boldsymbol{A}} < 0$ as the stable operating region when analyzing the generalization gap in the next experiment. These results are illustrated in Figure 1.

Experiment 2 (Length-Independent Generalization): Here we evaluate the generalization behavior across varying sequence lengths. Unlike Experiment 1, we initialize the models in a *marginally stable* regime with $s_{\boldsymbol{A}} = 0$, which results in smoother training. The models are trained and tested on sequences of different lengths, and we measure the generalization gap as the difference between training and test losses. As shown in Figure 2, this gap remains relatively stable across sequence lengths, with no consistent increasing or decreasing trend. These results support our theoretical claim that selective SSMs exhibit length-independent generalization behavior.

Interestingly, the mechanism behind this behavior is foreshadowed in Experiment 1: when initialized in an unstable regime, the model naturally pushes $s_{\boldsymbol{A}}$ below zero during training, moving toward

[2] Our code is available at https://github.com/Arya-Honarpisheh/gen_err_sel_ssm.

stability to enable learning. However, it stabilizes just enough to preserve rich temporal information. This suggests that **selective SSMs are implicitly biased toward operating near the stability boundary**, where they can extract long-range dependencies without incurring the exponential instability associated with longer sequences. This is a manifestation of the trade-off between expressivity and generalization. In Experiment 2, starting near this boundary allows the model to train smoothly, leading to consistent generalization across different sequence lengths.

6 Conclusion and Future Work

In this paper, we derived new generalization gap bounds for selective SSMs by leveraging their embedded linear-attention mechanism. As a corollary to our main result, we obtained a bound for linear attention, which illustrates the underlying connections explicitly. Our analysis revealed that the spectral abscissa $s_{\boldsymbol{A}}$ of the continuous-time state matrix $\boldsymbol{A}_c$ governs selective SSMs' generalization behavior: models with $s_{\boldsymbol{A}} < 0$ enjoy length-independent guarantees, while those with $s_{\boldsymbol{A}} > 0$ suffer exponential growth in error. Finally, our experiments supported these theoretical findings, showing that models satisfying $s_{\boldsymbol{A}} < 0$ indeed generalize well on long inputs. An important direction for future work is to improve these generalization error bounds. This could involve refining the analysis under alternative assumptions or exploring different techniques, such as directly bounding the Rademacher complexity instead of relying on covering-based arguments. Another promising avenue is to extend generalization analysis to other variants of deep architectures.

7 Limitations

This work considers a selective SSM as the discretization of a continuous-time state-space model, following the formulation used in Mamba. Hence, the theory developed here does not directly apply to other variants of selective SSMs where, for example, the matrix $A(\boldsymbol{u})$ exhibits a different dependency on the input. This work assumes that the training and test data are drawn from the same distribution. Out-of-distribution generalization of selective SSMs is not addressed. For a recent analysis of length generalization, a specific type of out-of-distribution setting where test sequences are longer than those seen during training, the interested reader is referred to Buitrago and Gu [41].

8 Acknowledgments

This work was partially supported by NSF grants CNS–2038493 and CMMI–2208182, AFOSR grant FA9550-19-1-0005, and ONR grant N00014-21-1-2431.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, page 6000–6010, 2017.

[2] 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. *Advances in neural information processing systems*, 34:572–585, 2021.

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

[4] Rudolph Emil Kalman et al. A new approach to linear filtering and prediction problems [j]. *Journal of basic Engineering*, 82(1):35–45, 1960.

[5] Dániel Rácz, Mihály Petreczky, and Bálint Daróczy. Length independent generalization bounds for deep ssm architectures. In *Next Generation of Sequence Modeling Architectures Workshop at ICML 2024*, 2024.

[6] Fusheng Liu and Qianxiao Li. From generalization analysis to optimization designs for state space models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

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

[8] Tri Dao and Albert Gu. Transformers are ssms: generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning*, pages 10041–10071. JMLR.org, 2024.

[9] Ameen Ali, Itamar Zimerman, and Lior Wolf. The hidden attention of mamba models. *arXiv preprint arXiv:2403.01590*, 2024.

[10] Annan Yu, Michael W Mahoney, and N Benjamin Erichson. Hope for a robust parameterization of long-memory state space models. In *The Thirteenth International Conference on Learning Representations*.

[11] Nicola Muca Cirone, Antonio Orvieto, Benjamin Walker, Cristopher Salvi, and Terry Lyons. Theoretical foundations of deep selective state-space models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[12] Yash Sarrof, Yana Veitsman, and Michael Hahn. The expressive capacity of state space models: A formal language perspective. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[13] Vinoth Nandakumar, Qiang Qu, Peng Mi, and Tongliang Liu. State space models can express n-gram languages. *Transactions on Machine Learning Research*, 2025.

[14] Aleksandar Terzic, Michael Hersche, Giacomo Camposampiero, Thomas Hofmann, Abu Sebastian, and Abbas Rahimi. On the expressiveness and length generalization of selective state space models on regular languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20876–20884, 2025.

[15] Annan Yu and N. Benjamin Erichson. Block-biased mamba for long-range sequence processing. In *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

[16] Zhiyang Wang, Juan Cerviño, and Alejandro Ribeiro. Generalization of graph neural networks is robust to model mismatch. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 21402–21410, 2025.

[17] Sanae Lotfi, Yilun Kuang, Marc Finzi, Brandon Amos, Micah Goldblum, and Andrew G Wilson. Unlocking tokens as data points for generalization bounds on larger language models. *Advances in Neural Information Processing Systems*, 37:9229–9256, 2024.

[18] Yunjuan Wang and Raman Arora. On the stability and generalization of meta-learning. *Advances in Neural Information Processing Systems*, 37:83665–83710, 2024.

[19] Jin Zhang, Ze Liu, Defu Lian, and Enhong Chen. Generalization error bounds for two-stage recommender systems with tree structure. *Advances in Neural Information Processing Systems*, 37:37070–37099, 2024.

[20] Yi-Fan Zhang and Min-Ling Zhang. Generalization analysis for label-specific representation learning. *Advances in Neural Information Processing Systems*, 37:104904–104933, 2024.

[21] Rayna Andreeva, Benjamin Dupuis, Rik Sarkar, Tolga Birdal, and Umut Simsekli. Topological generalization bounds for discrete-time stochastic optimization algorithms. *Advances in Neural Information Processing Systems*, 37:4765–4818, 2024.

[22] Kaibo Zhang, Yunjuan Wang, and Raman Arora. Stability and generalization of adversarial training for shallow neural networks with smooth activation. *Advances in Neural Information Processing Systems*, 37:16160–16193, 2024.

[23] Uday Kiran Reddy Tadipatri, Benjamin David Haeffele, Joshua Agterberg, and Rene Vidal. A convex relaxation approach to generalization analysis for parallel positively homogeneous networks. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025.

[24] Tong Zhang. Covering number bounds of certain regularized linear function classes. *Journal of Machine Learning Research*, 2(Mar):527–550, 2002.

[25] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

[26] Benjamin L Edelman, Surbhi Goel, Sham Kakade, and Cyril Zhang. Inductive biases and variable creation in self-attention mechanisms. In *International Conference on Machine Learning*, pages 5793–5831. PMLR, 2022.

[27] Jacob Trauger and Ambuj Tewari. Sequence length independent norm-based generalization bounds for transformers. In *International Conference on Artificial Intelligence and Statistics*, pages 1405–1413. PMLR, 2024.

[28] Lan V Truong. On rank-dependent generalisation error bounds for transformers. *arXiv preprint arXiv:2410.11500*, 2024.

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

[30] Richard M Dudley. The sizes of compact subsets of hilbert space and continuity of gaussian processes. *Journal of Functional Analysis*, 1(3):290–330, 1967.

[31] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[32] Minshuo Chen, Xingguo Li, and Tuo Zhao. On generalization bounds of a family of recurrent neural networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS), 2020*, 2020.

[33] Jiong Zhang, Qi Lei, and Inderjit Dhillon. Stabilizing gradients for deep neural networks via efficient svd parameterization. In *International Conference on Machine Learning*, pages 5806–5814. PMLR, 2018.

[34] Shida Wang and Qianxiao Li. StableSSM: Alleviating the curse of memory in state-space models through stable reparameterization. In *Forty-first International Conference on Machine Learning*, 2024.

[35] Fusheng Liu and Qianxiao Li. Autocorrelation matters: Understanding the role of initialization schemes for state space models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=sZJNkorXMk.

[36] Xuewei Cheng, Ke Huang, and Shujie Ma. Generalization and risk bounds for recurrent neural networks. *Neurocomputing*, page 128825, 2024.

[37] Yannik Keller. Solving vanishing gradients from model parameter collapse. https://yannikkeller.substack.com/p/solving-vanishing-gradients-from?r=3avwpj&triedRedirect=true, 2024.

[38] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150, 2011.

[39] Nikita Nangia and Samuel Bowman. Listops: A diagnostic dataset for latent tree learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 92–99, 2018.

[40] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena : A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2021.

[41] Ricardo Buitrago and Albert Gu. Understanding and improving length generalization in recurrent models. In *Forty-second International Conference on Machine Learning*.

[42] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations (ICLR), Conference Track Proceedings*, 2015.

[43] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[44] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[45] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[47] Jinyang Liu, Wondmgezahu Teshome, Sandesh Ghimire, Mario Sznaier, and Octavia Camps. Solving masked jigsaw puzzles with diffusion vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23009–23018, 2024.

[48] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. *Advances in neural information processing systems*, 33: 1474–1487, 2020.

[49] Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022.

[50] Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional language models. In *International Conference on Machine Learning*. PMLR, 2023.

[51] Pierre Alquier. User-friendly introduction to pac-bayes bounds. *Foundations and Trends® in Machine Learning*, 17(2):174–303, 2024. doi: 10.1561/2200000100.

[52] David A McAllester. Some pac-bayesian theorems. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 230–234, 1998.

[53] Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. *Advances in neural information processing systems*, 30, 2017.

[54] Thomas Steinke and Lydia Zakynthinou. Reasoning about generalization via conditional mutual information. In *Conference on Learning Theory*, pages 3437–3452. PMLR, 2020.

[55] Mahdi Haghifam, Jeffrey Negrea, Ashish Khisti, Daniel M Roy, and Gintare Karolina Dziugaite. Sharpened generalization bounds based on conditional mutual information and an application to noisy, iterative algorithms. *Advances in Neural Information Processing Systems*, 2020.

[56] Marek Karpinski and Angus Macintyre. Polynomial bounds for vc dimension of sigmoidal and general pfaffian neural networks. *Journal of Computer and System Sciences*, 54(1):169–176, 1997.

[57] Pascal Koiran and Eduardo D Sontag. Neural networks with quadratic vc dimension. *journal of computer and system sciences*, 54(1):190–198, 1997.

[58] Eduardo D Sontag. A learning result for continuous-time recurrent neural networks. *Systems & control letters*, 34(3):151–158, 1998.

[59] Eric Baum and David Haussler. What size net gives valid generalization? *Advances in neural information processing systems*, 1, 1988.

[60] Peter Bartlett, Vitaly Maiorov, and Ron Meir. Almost linear vc dimension bounds for piecewise polynomial networks. *Advances in neural information processing systems*, 11, 1998.

[61] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[62] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[63] P.L. Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory*, 44(2):525–536, 1998. doi: 10.1109/18.661502.

[64] Peter L Bartlett, Olivier Bousquet, and Shahar Mendelson. Local rademacher complexities. *The Annals of Statistics*, 33(4):1497–1537, 2005.

[65] Colin Wei and Tengyu Ma. Data-dependent sample complexity of deep neural networks via lipschitz augmentation. *Advances in neural information processing systems*, 32, 2019.

[66] Mehryar Mohri. Foundations of machine learning, 2018.

A Related Work

Self-attention is the core mechanism behind the Transformer architecture, introduced as an alternative to RNNs and CNNs for sequence processing [1, 42]. While the concept of attention predates the Transformer, its introduction marked a pivotal shift in the scale of large models. An attention mechanism assigns scores to each pair of elements in a sequence to measure their relevance to each other. The self-attention mechanism, drawing inspiration from the key–query analogy used in relational databases, captures dependencies between elements of an input sequence, where inputs attend to each other within the same sequence. Since their introduction, Transformers have been extensively studied and refined, leading to numerous variants, including sparse and low-rank adaptations and widespread applications across domains such as natural language processing [43, 44] and computer vision [45–47]. Related to our work is the connection between SSMs and attention [8, 9].

State-space models are a new class of foundation models, introduced by Gu et al. [2] as an alternative to Transformers for sequence processing. Rooted in the classical state-space representations introduced by Kalman et al. [4] in control theory, SSMs leverage state-space representations to efficiently model long-range dependencies in sequential data. The foundation of SSMs can be traced to the HiPPO framework, which established a mathematical basis for encoding and preserving long-range dependencies using orthogonal polynomial projections [48]. Building on this foundation, the first practical implementation of SSMs is the S4 model, which utilized HiPPO as an initialization scheme [3]. With the empirical success of S4 on the Long Range Arena benchmark [40], SSMs gained widespread attention, prompting several extensions and refinements. S4D simplified training with diagonal initializations [49], S5 introduced a multi-input multi-output structure for greater flexibility [29], and Hyena explored hierarchical convolutions [50]. Selective SSMs introduced in the Mamba model by Gu and Dao [7] extend LTI SSMs by using linear projections of the input to construct and discretize SSMs, resulting in a nonlinear time-variant architecture. These properties make selective SSMs closely resemble self-attention, as highlighted by Dao and Gu [8] while introducing Mamba-2.

Generalization bounds are central in the probably approximately correct (PAC) learning framework, which formalizes a model's ability to achieve low error on unseen data with high probability, provided sufficient training data. The PAC–Bayes framework provides probabilistic guarantees on generalization through a KL divergence between posterior and prior distributions [51, 52]. In the information-theoretic framework, the generalization gap is controlled by the mutual information between training data and model parameters [53–55]. Earlier studies explored statistical guarantees based on VC-dimension and shattering bounds extensively [56–60]. A related line of work uses Rademacher complexity to control the generalization gap, often through chaining and Dudley's integral, leading to margin- or norm-based bounds [30, 61–63, 25]. This framework was later refined through local Rademacher complexity [64], which focuses on subsets of the hypothesis class near the empirical minimizer, yielding sharper bounds for deep models [65]. The recent works on norm-based generalization bounds utilize covering numbers, a fundamental tool for bounding the capacity of function classes [61]. Zhang [24] laid the groundwork for understanding the capacity of regularized linear function classes through covering numbers. Later, Bartlett et al. [25] established generalization bounds for neural networks using covering numbers based on the work of Zhang [24]. These methods have been extended to Transformers in recent studies [26–28], where different aspects such as length or rank dependency have been emphasized. For LTI SSMs, the works of Rácz et al. [5], Liu and Li [6] draw inspiration from this line of research but primarily leverage the structure of LTI systems to derive their bounds from a state-space perspective.

B Experimental Details

In both experiments, we employ an embedding layer implemented using `torch.nn.Embedding`, which maps input tokens into a continuous vector space. This is followed by a selective SSM block, configured with $N = 4$ states per channel and $d = 16$ channels. The selective SSM block is parameterized as

$$\Theta_{\text{SSM}} = \{\boldsymbol{A}_c, \boldsymbol{W}_{\boldsymbol{B}}, \boldsymbol{W}_{\boldsymbol{C}}, p, q, w\}$$

where each component is defined in Section 2.2. The matrix $\boldsymbol{A}_c \in \mathbb{R}^{Nd\times Nd}$ represents the state matrices across channels. In the code implementation, we store $\boldsymbol{A}_c$ structured as a $\mathbb{R}^{d\times N}$ matrix where each row of $\boldsymbol{A}_c$ corresponds to the diagonal elements of a distinct diagonal state matrix $\boldsymbol{A}_c^{(j)} \in \mathbb{R}^{N\times N}$ for channel j, where $j \in \{1, \ldots, d\}$. This parameterization follows the official

implementation of Mamba [7], ensuring computational efficiency while maintaining expressive capacity. The remaining parameters in Θ_{SSM} have the exact dimensions described in Section 2.2: $\boldsymbol{W_B}, \boldsymbol{W_C} \in \mathbb{R}^{N\times d}$, $q \in \mathbb{R}^d$, and $p \in \mathbb{R}$. The first experiment investigates how the stability margin affects training, particularly showing that the initial loss grows exponentially with sequence length T when $s_{\boldsymbol{A}} > 0$. To focus on early training dynamics, we train for only 10 epochs using 10% of the dataset, balanced across labels, to control loss magnitude and avoid label bias. In contrast, the second experiment uses the full datasets to train models to convergence and evaluate the generalization gap. The second experiment is repeated five times with different random seeds. For very long sequences in the IMDb dataset, even with $s_{\boldsymbol{A}} = 0$ initialization, training may fail even at the very first stage, preventing the model from stabilizing during the first few epochs. In such cases, we retry with a different seed that yields a non-NaN value in the first epoch. The final results are reported as the mean over five successful runs $\pm$ one standard deviation.

B.1 Majority Dataset

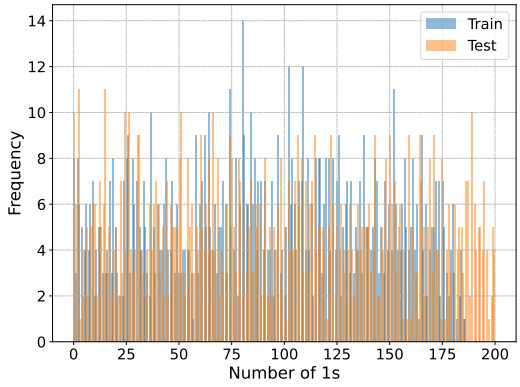

Figure 3: **Majority**. Histogram of ones, $m = 1000$ samples each for train and test, sequence length $T = 200$.

Majority is a synthetic dataset similar to the dataset used in the experiments of Trauger and Tewari [27], but with modifications. Each sample consists of a sequence of ones and zeros, forming the basis of a binary classification task. The class label indicates whether a sequence contains more ones than zeros. A sample sequence u_1 with $T = 20$ and its label z_1 would be as the following:

$$u_1 = [1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1]$$
$$z_1 = 1$$

Since the task involves only two unique elements, the vocabulary size is set to 2, and each element in the sequence is projected into embeddings of dimension d when they pass the embedding layer. Both the training and test sets contain $m = 1000$ samples. To ensure a uniform distribution of ones and zeros across sequence lengths, we generate sequences such that the number of ones varies approximately evenly from 0 to T. To introduce some imbalance, we modify the training set by randomly flipping 10% of ones to zeros after generating the sequences and labels. As shown in Figure 3, this results in a noticeable reduction in sequences with a high number of ones. Specifically, towards the maximum sequence length T, fewer samples retain exactly T ones due to these perturbations, altering the original distribution.

B.2 IMDb Large Movie Review Dataset

IMDb large movie review dataset [38] is a standard benchmark for sentiment analysis models and part of the Long-Range Arena (LRA) benchmark [40]. The dataset contains 50,000 movie reviews, evenly split between positive and negative labels, and is divided into training and test sets of 25,000 reviews each. The task is binary sentiment classification, aiming to predict whether a review expresses a positive or negative sentiment. The dataset's balanced nature ensures unbiased model evaluation.

We chose IMDb for its diverse sequence lengths, as shown in Table 2 and Figure 4. To train effectively, we used the entire dataset, truncating or padding sequences to a fixed length. For our experiments, we

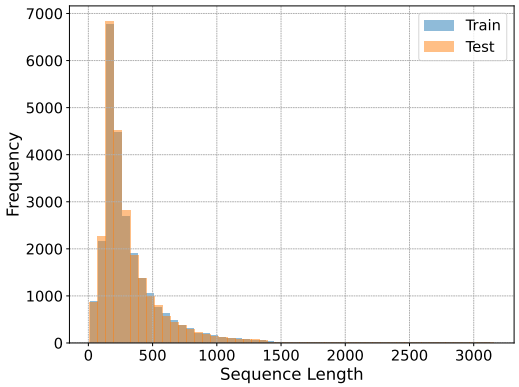

Figure 4: **IMDb**. Histogram of sequence lengths for both the training and test splits.

chose sequence lengths between 100 and 2000 tokens, based on the distribution observed in Figure 4. As shown in Figure 2 (bottom), test accuracy increases from 100 to 300 tokens, then stabilizes. The generalization gap, visible in the bottom plot, reflects this trend. The average sequence length is 314 tokens, with many sequences exceeding 300 tokens (Figure 4). Truncating sequences longer than 300 tokens results in the loss of valuable information, potentially reducing predictive accuracy, as demonstrated by the following examples.

Short Sample
Text: "I don't know why I like this movie so well, but I never get tired of watching it."
Label: Positive (1)
Length: 24

Long Sample
Text: "This movie was recently released on DVD in the US and I finally got the chance..."
Label: Negative (0)
Length: 1833

For shorter sequences, key indicators of the sentiment label often appear early in the text, making it easier for the model to make predictions. However, for longer sequences, these indicators may not be immediately apparent, as the sentiment may be spread across the entire review. In such cases, retaining the full context of the sequence becomes crucial for accurate prediction. This is particularly evident in the test loss observed in the bottom Figure 2, where truncating longer sequences results in a loss of critical context, reducing the model's accuracy.

	Max	Min	Average	Median
Train	3127	13	314	233
Test	3157	10	307	230

Table 2: **IMDb**. Sequence length details for training and test splits.

B.3 ListOps Dataset

ListOps is a synthetic benchmark designed to evaluate a model's ability to perform hierarchical reasoning over long sequences first introduced in Nangia and Bowman [39]. Each sample in this dataset is a bracketed expression consisting of nested mathematical operations, such as

$$[\text{MED } 4\ 8\ [\text{MIN } 7\ 2]\ 2\ 3],$$

which evaluates to a single-digit integer. The correct label depends on the complete nested structure, making the task dependent on the entire length of the sequence. We adopt the preprocessed version provided by Tay et al. [40]. The vocabulary consists of digits, "[","]","(",")", and operator tokens (`MAX`, `MIN`, `MED`, `SUM`). Sequence lengths are set between $T = 100$ and $T = 1000$ in increments of 100. For each sequence length, we generate data in the range $[100k - 5,\ 100k + 5]$.

B.4 Experiment 1: Bounded Parameter Norms

The following table provides an example training log from Experiment 1 (IMDb, $T = 350$). The spectral abscissa $s_{\boldsymbol{A}}$ is already plotted in Figure 1 (bottom middle, orange line). At the beginning of a stable training cycle, in which $s_{\boldsymbol{A}}$ is successfully pushed toward 0 from above and the loss drops significantly within 10 epochs, the parameter norms do not show strictly increasing trends. This supports the claim that these norms remain bounded. Additional logs are available in our GitHub repository.

epoch	$s_{\boldsymbol{A}}$	$\lvert p\rvert$	$\lVert q\rVert_2$	$\lVert W_B\rVert_2$	$\lVert W_B\rVert_{1,1}$	$\lVert W_C\rVert_2$	$\lVert W_C\rVert_{1,1}$	$\lVert A_c\rVert_2$	$\max \lVert u[t]\rVert_2$	**Loss**
0	0.100	1.43	4.12	7.68	51.0	7.40	48.5	9.51	7.39	-
1	0.069	1.46	3.89	7.12	45.9	6.85	43.4	9.41	7.07	6.4×10^{20}
3	0.005	1.62	3.56	6.53	41.4	6.33	39.1	9.21	6.49	347.7
5	-0.006	1.58	3.44	6.21	39.5	6.15	39.0	9.01	6.19	2.04
7	-0.009	1.58	3.37	6.05	38.6	6.13	39.0	8.84	6.00	1.00
9	-0.010	1.61	3.33	5.91	37.5	6.12	39.0	8.66	5.82	0.67

Table 3: **IMDb**. Parameter logs during training epochs for Experiment 1, $T = 350$.

B.5 Experiment 1: Sweeping Spectral Abscissa with Constant Sequence Length

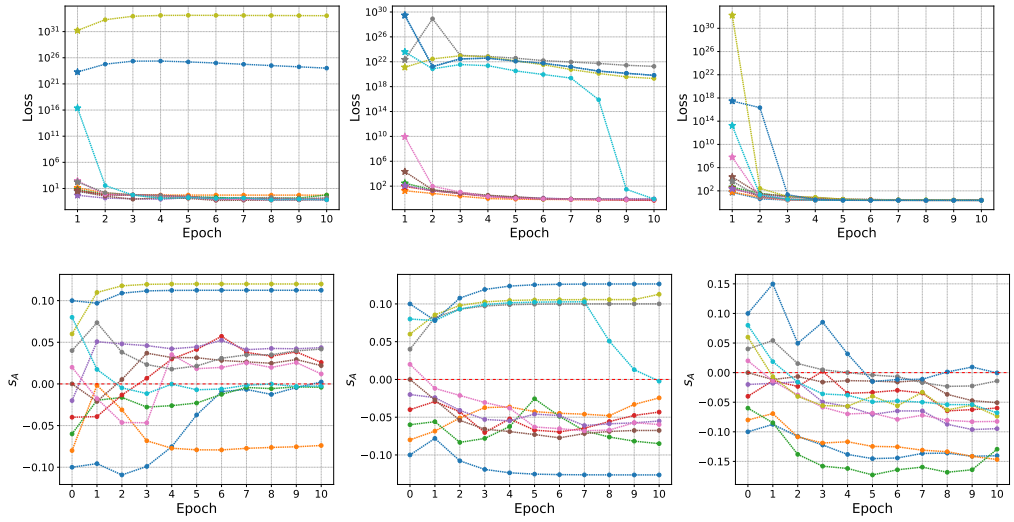

Figure 5: **Experiment 1. Top:** Training loss vs epochs for **Left:** Majority, $T = 250$, **Middle:** IMDb, $T = 500$, **Right:** ListOps, $T = 300$. **Bottom:** Evolution of $s_{\boldsymbol{A}}$ vs epochs for the same datasets. We sweep the $s_{\boldsymbol{A}}$ values from -0.1 to 0.1 in 0.02 increments.

To understand the effect of different initialization regimes in more detail, we conduct Experiment 1 again, but this time we sweep the $s_{\boldsymbol{A}}$ values while keeping T fixed. In order to observe the full effect without encountering runs that diverge (i.e., contain NaN losses), we choose $T = 250$ for Majority, $T = 500$ for IMDb, and $T = 300$ for ListOps. The results in Figure 5 are consistent with those in Figure 1. Unstable initializations (where $s_{\boldsymbol{A}} > 0$) often lead to training failure due to exploding losses, as other parameters have not yet adapted to stabilize the system. In cases where training succeeds, $s_{\boldsymbol{A}}$ is quickly pushed toward zero from above, restoring stability. In contrast, stable initializations ($s_{\boldsymbol{A}} < 0$) enable smooth training and remain stable throughout for IMDb and ListOps.

For Majority, stable initializations below zero are also pushed toward zero, and some even end up slightly above it (around $s_{\boldsymbol{A}} \approx 0.05$), which is acceptable since other parameters ($\mathfrak{B}_q$, $\mathfrak{B}_u$, and p) can compensate according to (15). The key insight is that initialization determines whether training can proceed stably. When $s_{\boldsymbol{A}}$ starts in an unstable region, large initial losses arise before other parameters can adapt, often causing divergence. In contrast, initializing in the stable regime enables smooth optimization and coordinated parameter adaptation, allowing $\rho_{\boldsymbol{A}}$ to remain balanced through (15). Thus, proper initialization of $s_{\boldsymbol{A}}$ is essential for both convergence and stable expressivity.

C Useful Lemmas

Lemma C.1. *Let $\boldsymbol{B}_c = \mathbf{I}_d \otimes \boldsymbol{W_B} u[t]$. Then, $\|\boldsymbol{B}_c\|_2 \leq \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u$.*

Proof.

$$\|\boldsymbol{B}_c\|_2^2 = \|\mathbf{I}_d \otimes \boldsymbol{W_B} u[t]\|_2^2 = \|\boldsymbol{W_B} u[t]\|_2^2 \leq \|\boldsymbol{W_B}\|_2^2 \mathfrak{B}_u^2 \leq \mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_u^2. \tag{21}$$

□

Lemma C.2. *Let $\boldsymbol{C}_c = \mathbf{I}_d \otimes u[t]^\top \boldsymbol{W}_{\boldsymbol{C}}^\top$. Then, $\|\boldsymbol{C}_c\|_2 \leq \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u$.*

Proof.

$$\|\boldsymbol{C}_c\|_2^2 = \|\mathbf{I}_d \otimes u[t]^\top \boldsymbol{W}_{\boldsymbol{C}}^\top\|_2^2 = \|u[t]^\top \boldsymbol{W}_{\boldsymbol{C}}^\top\|_2^2 = \|\boldsymbol{W_C} u[t]\|_2^2 \leq \|\boldsymbol{W_C}\|_2^2 \mathfrak{B}_u^2 \leq \mathfrak{B}_{\boldsymbol{C}}^2 \mathfrak{B}_u^2. \tag{22}$$

□

Lemma C.3. *Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be matrices such that $\|e^{\boldsymbol{X}}\|_2 \leq \rho$ and $\|e^{\boldsymbol{Y}}\|_2 \leq \rho$. Then,*

$$\|e^{\boldsymbol{X}} - e^{\boldsymbol{Y}}\|_2 \leq \rho \|\boldsymbol{Y} - \boldsymbol{X}\|_2. \tag{23}$$

Proof. Using the fundamental theorem of calculus, we express the difference as

$$e^{\boldsymbol{X}} - e^{\boldsymbol{Y}} = \int_0^1 e^{t\boldsymbol{Y} + (1-t)\boldsymbol{X}} (\boldsymbol{Y} - \boldsymbol{X})\, dt. \tag{24}$$

Taking the spectral norm on both sides and applying submultiplicativity, we obtain

$$\begin{aligned} \|e^{\boldsymbol{X}} - e^{\boldsymbol{Y}}\|_2 &\leq \left\| \int_0^1 e^{t\boldsymbol{Y} + (1-t)\boldsymbol{X}} (\boldsymbol{Y} - \boldsymbol{X})\, dt \right\|_2 \\ &\leq \int_0^1 \|e^{t\boldsymbol{Y} + (1-t)\boldsymbol{X}} (\boldsymbol{Y} - \boldsymbol{X})\|_2\, dt \\ &\leq \|\boldsymbol{Y} - \boldsymbol{X}\|_2 \int_0^1 \|e^{t\boldsymbol{Y} + (1-t)\boldsymbol{X}}\|_2\, dt \\ &\leq \rho \|\boldsymbol{Y} - \boldsymbol{X}\|_2. \end{aligned} \tag{25}$$

□

Lemma C.4. *Let $\hat{p}$ be an ϵ_p-cover for p and $\hat{q}$ be an ϵ_q-cover for q. Then,*

$$|\Delta[t] - \hat{\Delta}[t]| \leq \epsilon_p + \epsilon_q. \tag{26}$$

Proof. The derivative of the soft plus function, $\ln(1 + e^x)$, is $\frac{e^x}{1+e^x}$, which is bounded by 1. Thus, $\ln(1 + e^x)$ is 1-Lipschitz. using this property, we obtain

$$\begin{aligned} |\Delta[t] - \hat{\Delta}[t]| &= \left| \ln(1 + e^{p + q^\top u[t]}) - \ln(1 + e^{\hat{p} + \hat{q}^\top u[t]}) \right| \\ &\leq \left| p - \hat{p} + (q - \hat{q})^\top u[t] \right| \\ &\leq |p - \hat{p}| + \|(q - \hat{q})^\top u[t]\|_2. \end{aligned} \tag{27}$$

□

Lemma C.5. *Let $\hat{w}$ be an ϵ_w-cover for w. Then,*

$$\left| w^\top y[T] - \hat{w}^\top \hat{y}[T] \right| \leq \mathfrak{B}_w \|y[T] - \hat{y}[T]\|_2 + \epsilon_w. \tag{28}$$

Proof. Rewriting the LHS, we obtain

$$\left| w^\top y[T] - \hat{w}^\top \hat{y}[T] \right| = \left| w^\top (y[T] - \hat{y}[T]) + (w - \hat{w})^\top \hat{y}[T] \right|. \tag{29}$$

Applying triangle inequality results in

$$\begin{aligned} \left| w^\top y[T] - \hat{w}^\top \hat{y}[T] \right| &\leq \|w\|_2 \|y[T] - \hat{y}[T]\|_2 + \left| (w - \hat{w})^\top \hat{y}[T] \right| \\ &\leq \mathfrak{B}_w \|y[T] - \hat{y}[T]\|_2 + \epsilon_w. \end{aligned} \tag{30}$$

□

Lemma C.6. *Let $\hat{\boldsymbol{C}}_c$ be an ϵ_C-cover for $\boldsymbol{C}_c$. Then,*

$$\|\boldsymbol{C}_c x[T] - \hat{\boldsymbol{C}}_c \hat{x}[T]\|_2 \leq \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u \|x[T] - \hat{x}[T]\|_2 + \epsilon_C. \tag{31}$$

Proof. The LHS can be bounded as follows:

$$\begin{aligned} &\leq \|\boldsymbol{C}_c \left(x[T] - \hat{x}[T]\right) + (\boldsymbol{C}_c - \hat{\boldsymbol{C}}_c)\hat{x}[T]\|_2 \\ &\leq \|\boldsymbol{C}_c\|_2 \|x[T] - \hat{x}[T]\|_2 + \|(\boldsymbol{C}_c - \hat{\boldsymbol{C}}_c)\hat{x}[T]\|_2. \end{aligned} \tag{32}$$

Applying Lemma (C.2) completes the proof. □

Lemma C.7. *Let $\hat{\boldsymbol{W}}_B$ be a cover for $\boldsymbol{W}_B$. Then, for any $v \in \mathbb{R}^d$ such that $\|v\|_2 = \mathfrak{B}_v$, we obtain*

$$\|(\boldsymbol{B}_c - \hat{\boldsymbol{B}}_c)v\|_2 \leq \mathfrak{B}_v \|(\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)u\|_2. \tag{33}$$

Proof. We use the Kronecker product property $(\boldsymbol{X} \otimes \boldsymbol{Y})\,\mathrm{vec}(\boldsymbol{V}) = \mathrm{vec}\left(\boldsymbol{Y}\boldsymbol{V}\boldsymbol{X}^T\right)$. Take $\boldsymbol{X}$ as $\mathbf{I}_d$, $\boldsymbol{V}$ as $v^\top$, and $\boldsymbol{Y}$ as $(\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)u$ to obtain

$$\begin{aligned} \|(\boldsymbol{B}_c - \hat{\boldsymbol{B}}_c)v\|_2 &= \| \left(\mathbf{I}_d \otimes (\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)u\right) v\|_2 \\ &= \| \,\mathrm{vec}\left((\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)uv^\top\right) \|_2. \end{aligned} \tag{34}$$

From the definition of the Frobenius norm, we obtain

$$\begin{aligned} \|(\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)uv^\top\|_F &\leq \|(\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)u\|_F \|v^\top\|_F \\ &= \|(\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)u\|_2 \|v\|_2 \\ &\leq \mathfrak{B}_v \|(\boldsymbol{W}_B - \hat{\boldsymbol{W}}_B)u\|_2. \end{aligned} \tag{35}$$

□

Lemma C.8. *Let $\hat{\boldsymbol{W}}_C$ be a cover for $\boldsymbol{W}_C$. Then, for any $v \in \mathbb{R}^{Nd}$ such that $\|v\|_2 = \mathfrak{B}_v$, we obtain*

$$\|(\boldsymbol{C}_c - \hat{\boldsymbol{C}}_c)v\|_2 \leq \mathfrak{B}_v \|(\boldsymbol{W}_C - \hat{\boldsymbol{W}}_C)u\|_2. \tag{36}$$

Proof. Similar to Lemma C.7. □

Lemma C.9 (Edelman et al. [26], Lemma A.8)**.** *For $\alpha_i, \beta_i \geq 0$, the solution to the following optimization*

$$\begin{aligned} &\min_{\epsilon_1,\ldots,\epsilon_n} \sum_{i=1}^{n} \frac{\alpha_i}{\epsilon_i^2} \\ &\textit{subject to} \sum_{i=1}^{n} \beta_i \epsilon_i = \epsilon \end{aligned} \tag{37}$$

is $\frac{\gamma^3}{\epsilon^2}$ and is achieved at $\epsilon_i = \frac{\epsilon}{\gamma}\left(\frac{\alpha_i}{\beta_i}\right)^{1/3}$, where $\gamma = \sum_{i=1}^n \alpha_i^{1/3}\beta_i^{\frac{2}{3}}$.

D Covering Numbers

Lemma D.1 (Bartlett et al. [25], Lemma 3.2)**.** *Let conjugate exponents (p, q) and (r, s) be given with $p \leq 2$, as well as positive reals (a, b, ϵ) and positive integer d_3. Let matrix $X \in \mathbb{R}^{d_1 \times d_2}$ be given with $\|X\|_{p,p} \leq b$. Then,*

$$\ln \mathcal{N}\left(\left\{XA : A \in \mathbb{R}^{d_2 \times d_3}, \|A\|_{q,s} \leq a\right\}, \epsilon, \|\cdot\|_F\right) \leq \left\lceil \frac{a^2 b^2 d_3^{2/r}}{\epsilon^2} \right\rceil \ln(2d_2 d_3). \tag{38}$$

Lemma D.2. *Let $\mathcal{F}_{\boldsymbol{A}_c} = \{\boldsymbol{A}_c \in \mathbb{R}^{Nd \times Nd} : \|\boldsymbol{A}_c\|_2 \leq \mathfrak{B}_{\boldsymbol{A}}$ and $\|\boldsymbol{A}_c\|_{2,1} \leq \mathfrak{M}_{\boldsymbol{A}}\}$. Then,*

$$\ln \mathcal{N}(\mathcal{F}_{A_c}, \epsilon_A, \|\cdot\|_2) \leq \frac{2\mathfrak{M}_A^2 Nd}{\epsilon_A^2} \ln(\sqrt{2}Nd). \tag{39}$$

Proof. Note that every ϵ_A-covering number for the Frobenius norm is also an ϵ_A-covering number for the spectral norm, as $\|\boldsymbol{A}-\hat{\boldsymbol{A}}\|_2 \le \|\boldsymbol{A}-\hat{\boldsymbol{A}}\|_F \le \epsilon_A$. Therefore,

$$\begin{aligned}\ln \mathcal{N}(\mathcal{F}_{A_c}, \epsilon_A, \|\cdot\|_2) &\le \ln \mathcal{N}(\mathcal{F}_{A_c}, \epsilon_A, \|\cdot\|_F) \\ &\le \ln \mathcal{N}(\{\boldsymbol{A}_c \in \mathbb{R}^{Nd\times Nd} : \|\boldsymbol{A}_c\|_{2,1} \le \mathfrak{M}_{\boldsymbol{A}}\}, \epsilon_A, \|\cdot\|_F).\end{aligned} \tag{40}$$

Thus, we instantiate Lemma D.1 with $p=q=2$ and $s=1, r=\infty$. Take X to be identity and thus $b=\sqrt{Nd}$ which results in

$$\ln \mathcal{N}(\{\boldsymbol{A}_c \in \mathbb{R}^{Nd\times Nd} : \|\boldsymbol{A}_c\|_{2,1} \le \mathfrak{M}_{\boldsymbol{A}}\}, \epsilon_A, \|\cdot\|_F) \le \left\lceil \frac{\mathfrak{M}_A^2(\sqrt{Nd})^2}{\epsilon_A^2} \right\rceil \ln(2NdNd). \tag{41}$$

□

Lemma D.3 (Trauger and Tewari [27], Lemma 3.6)**.** *Let* $m \ge d_2$, $\mathcal{F}_W = \{\boldsymbol{W}u : \boldsymbol{W} \in \mathbb{R}^{d_1\times d_2}, \|\boldsymbol{W}\|_{1,1} \le \mathfrak{M}_W\}$. *If* $\|u\|_2 \le \mathfrak{B}_u$, *then*

$$\ln \mathcal{N}_\infty(\mathcal{F}_W, \epsilon_W, \|\cdot\|_2) \le \frac{\mathfrak{B}_u^2\mathfrak{M}_W^2}{\epsilon_W^2}\ln(2d_1d_2+1). \tag{42}$$

Remark. The removal of the dependency on m in the log covering number for a function class is nontrivial and requires specific assumptions about the norm bounds. For similar log covering bounds that are independent of m, refer to [27] and the lemmas therein.

Lemma D.4. *Let* $\mathcal{F}$ *be a function class such that* $\ln \mathcal{N}_\infty(\mathcal{F}, \epsilon, \|\cdot\|_2) \le \frac{\mathcal{C}_\mathcal{F}^2}{\epsilon^2}$ *and let* S *be the training set* $\{u_{(i)}, z_{(i)}\}_{i=1}^m$. *Assume the loss function* $l : \mathcal{Z}\times\mathcal{Z} \to \mathbb{R}$ *is upper bounded by the constant* $\mathfrak{c}_l$ *and Lipschitz continuous with constant* $\mathfrak{l}_l$. *Then, with probability at least* $1-\delta$,

$$\left|\mathbb{E}_{u,z}(l(h(u),z)) - \frac{1}{m}\sum_{i=1}^m l\left(h(u_{(i)}), z_{(i)}\right)\right| \le \frac{12\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\sqrt{m}}\left(1+\ln\left(\frac{\mathfrak{c}_l\sqrt{m}}{3\mathcal{C}_\mathcal{F}}\right)\right) + 3\mathfrak{c}_l\sqrt{\frac{\ln\left(\frac{2}{\delta}\right)}{2m}}. \tag{43}$$

Proof. By Theorem 3.2, and the fact that $\ln \mathcal{N}_2(l\circ\mathcal{F}, \epsilon, \|\cdot\|_2) \le \ln \mathcal{N}_\infty(l\circ\mathcal{F}, \epsilon, \|\cdot\|_2)$ (check Definition 1 in [24]), we have

$$\mathrm{Rad}(l\circ\mathcal{F}, S) \le \inf_{\alpha>0}\left(4\alpha + 12\int_\alpha^{\mathfrak{c}_l}\sqrt{\frac{\ln \mathcal{N}_\infty(l\circ\mathcal{F}, \epsilon, \|\cdot\|_2)}{m}}\,d\epsilon\right). \tag{44}$$

Upper bound $\ln \mathcal{N}_\infty(\mathcal{F}, \epsilon, m, \|\cdot\|_2)$ as specified by the lemma to obtain

$$\le \inf_{\alpha>0}\left(4\alpha + \frac{12}{\sqrt{m}}\int_\alpha^{\mathfrak{c}_l}\frac{\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\epsilon}\,d\epsilon\right) = \inf_{\alpha>0}\left(4\alpha + \frac{12\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\sqrt{m}}\ln\left(\frac{\mathfrak{c}_l}{\alpha}\right)\right) \tag{45}$$

in which we used $\ln \mathcal{N}_\infty(l\circ\mathcal{F}, \epsilon, \|\cdot\|_2) \le \mathfrak{l}_l \ln \mathcal{N}_\infty(\mathcal{F}, \epsilon, \|\cdot\|_2)$. The minimum of (45) occurs at $\alpha = \frac{3\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\sqrt{m}}$. Thus,

$$\le \frac{12\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\sqrt{m}} + \frac{12\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\sqrt{m}}\ln\left(\frac{\mathfrak{c}_l\sqrt{m}}{3\mathcal{C}_\mathcal{F}}\right) = \frac{12\mathfrak{l}_l\mathcal{C}_\mathcal{F}}{\sqrt{m}}\left(1+\ln\left(\frac{\mathfrak{c}_l\sqrt{m}}{3\mathcal{C}_\mathcal{F}}\right)\right). \tag{46}$$

Combining this bound on the Rademacher complexity with Theorem E.2 concludes the proof. □

E Proof for Theorem 3.3: Generalization Error Bound for Selective SSMs

Before presenting the proof of Theorem 3.3, we provide preliminary background, including the definition of Rademacher complexity and a standard theorem establishing its connection to the generalization gap. Then, we introduce four intermediate lemmas tailored to the selective SSMs. The first lemma establishes an upper bound on the spectral norm of the state matrix after t repetitions, namely $\|\boldsymbol{A}^t\|_2$. The second lemma bounds the distance between the time-varying product $\boldsymbol{A}^t$ and its corresponding cover, accounting for the input-dependent nature of the state matrices. These two lemmas are necessary and specific to selective SSMs, since, unlike standard RNNs, the state matrix

$\boldsymbol{A}$ is not fixed. As a result, classical norm bounds do not directly apply, and we must instead derive upper bounds in terms of the model parameters $\boldsymbol{A}_c$, p, and q, which govern the input-dependent dynamics. The third and fourth lemmas build upon the first two to inductively bound the distance between the output of a selective SSM and that of its corresponding cover. These results culminate in the proof of Theorem 3.3 provided at the end of this section.

Definition E.1 (**Rademacher complexity**). *For a given real-valued Function class $\mathcal{F}$ and a set of vectors $S = \{u_{(i)}\}_{i=1}^m$, the empirical Rademacher complexity is*

$$\mathrm{Rad}(\mathcal{F}, S) = \frac{1}{m}\mathbb{E}_\sigma \left(\sup_{f\in\mathcal{F}} \sum_{i=1}^{m} \sigma_i f(u_{(i)}) \right) \tag{47}$$

where $\sigma_i \in \{-1, 1\}$ are uniformly distributed i.i.d Rademacher random variables.

Theorem E.2 (Mohri [66], Theorem 3.3). *Let $\mathcal{F}$ be a hypothsis class $\{f : \mathcal{U} \to \mathcal{Z}\}$, and S be the training set $\{u_{(i)}, z_{(i)}\}_{i=1}^m$. Assume the loss function $l : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ is upper bounded by the constant $\mathfrak{c}_l$. Then, with probability more than $1 - \delta$*

$$\left| \mathbb{E}_{u,z}(l(f(u), z)) - \frac{1}{m}\sum_{i=1}^{m} l\left(f(u_{(i)}), z_{(i)}\right) \right| \le 2\,\mathrm{Rad}(l \circ \mathcal{F}, S) + 3\mathfrak{c}_l \sqrt{\frac{\ln(\frac{2}{\delta})}{2m}}. \tag{48}$$

Lemma E.3. *Let $s_{\boldsymbol{A}}$ be the spectral abscissa of $\boldsymbol{A}_c$ i.e. $\max_i \Re(\lambda_i(\boldsymbol{A}_c))$. Suppose $u[t] \le \mathfrak{B}_u$ and $\|q\|_2 \le \mathfrak{B}_q$. Then, given any arbitrary small $\eta > 0$, there exists a sufficiently large t such that*

$$\|\boldsymbol{A}^t\|_2 \le \rho_{\boldsymbol{A}}^t, \tag{49}$$

where

$$\rho_{\boldsymbol{A}} = \left(1 + e^{p - \mathfrak{B}_q\mathfrak{B}_u}\right)^{s_{\boldsymbol{A}}+\eta}. \tag{50}$$

Proof. From (9), we have

$$\|\boldsymbol{A}^t\|_2 = \left\| \prod_{j=T-t}^{T-1} e^{\Delta[j]\boldsymbol{A}_c} \right\|_2. \tag{51}$$

Given the assumptions of the lemma, and noting that the softplus function, $\ln(1 + e^x)$, is increasing, we derive the following lower bound:

$$\Delta[j] \ge \ln(1 + e^{p - \mathfrak{B}_q\mathfrak{B}_u}). \tag{52}$$

Since the spectral abscissa of $\boldsymbol{A}_c$ is $s_{\boldsymbol{A}}$, the spectral radius of $e^{\boldsymbol{A}_c}$ would be $e^{s_{\boldsymbol{A}}}$. By Gelfand's formula (Corollary 5.6.14 in [31]), we have that $\|(e^{\boldsymbol{A}_c})^t\|_2^{1/t} \to e^{s_{\boldsymbol{A}}}$ as $t \to \infty$. Consequently, for an arbitrary small positive number $\eta > 0$, there exists a sufficiently large t_0 such that for all $t \ge t_0$, the following bound holds:

$$\|(e^{\boldsymbol{A}_c})^t\|_2 \le \left(e^{s_{\boldsymbol{A}}+\eta}\right)^t.$$

This yields the desired exponential norm bound for large t:

$$\begin{aligned} \|\boldsymbol{A}^t\|_2 &= \left\| \left(e^{\boldsymbol{A}_c}\right)^{\sum_{j=T-t}^{T-1}\Delta[j]} \right\|_2 \\ &\le \left(e^{s_{\boldsymbol{A}}+\eta}\right)^{t\ln\left(1+e^{p-\mathfrak{B}_q\mathfrak{B}_u}\right)} \\ &= \left(1 + e^{p-\mathfrak{B}_q\mathfrak{B}_u}\right)^{(s_{\boldsymbol{A}}+\eta)t} = \rho_{\boldsymbol{A}}^t. \end{aligned} \tag{53}$$

□

Lemma E.4. *We have the following upper bound on $x[T]$:*

$$\|x[T]\|_2 \le \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 \frac{1 - \rho_A^T}{1 - \rho_A} \tag{54}$$

in which $\rho_{\boldsymbol{A}} = \left(1 + e^{p-\mathfrak{B}_q\mathfrak{B}_u}\right)^{s_A+\eta}$ and $\mathfrak{M}_\Delta = \ln(1 + e^{p+\mathfrak{B}_q\mathfrak{B}_u})$.

Proof.

$$\begin{aligned}\|x[T]\|_2 &\leq \left\|\sum_{t=0}^{T-1}\Big(\boldsymbol{A}^t\Delta[T-1-t]\big(\mathbf{I}_d\otimes \boldsymbol{W_B}u[T-1-t]\big)u[T-1-t]\Big)\right\|_2 \\ &\leq \sum_{t=0}^{T-1}\Big(\|\boldsymbol{A}^t\|_2\|\Delta[T-1-t]\big(\mathbf{I}_d\otimes \boldsymbol{W_B}u[T-1-t]\big)\|_2\|u[T-1-t]\|_2\Big) \\ &\leq \mathfrak{M}_\Delta\mathfrak{B}_u\sum_{t=0}^{T-1}\Big(\|\boldsymbol{A}^t\|_2\|\mathbf{I}_d\otimes \boldsymbol{W_B}u[T-1-t]\|_2\Big) \\ &\leq \mathfrak{M}_\Delta\mathfrak{B}_{\boldsymbol{B}}\mathfrak{B}_u^2\sum_{t=0}^{T-1}\rho_{\boldsymbol{A}}^t = \mathfrak{M}_\Delta\mathfrak{B}_{\boldsymbol{B}}\mathfrak{B}_u^2\frac{1-\rho_A^T}{1-\rho_A},\end{aligned} \tag{55}$$

where we used Lemma E.3 to bound $\|\boldsymbol{A}^t\|_2$ and Lemma C.1 to upper bound the term involving the Kronecker product to derive the last inequality. □

Lemma E.5. *Let $\hat{\boldsymbol{A}}_c$ be an ϵ_A-cover for $\boldsymbol{A}_c$, $\hat{p}$ be an ϵ_p-cover for p, and $\hat{q}$ be an ϵ_q-cover for q. Then,*

$$\|\boldsymbol{A}^t-\hat{\boldsymbol{A}}^t\|_2 \leq t\rho_{\boldsymbol{A}}^t(\mathfrak{M}_\Delta\epsilon_A+\mathfrak{B}_{\boldsymbol{A}}\epsilon_\Delta) \tag{56}$$

where $\mathfrak{M}_\Delta = \ln(1+e^{p+\mathfrak{B}_q\mathfrak{B}_u})$.

Proof. We start with

$$\begin{aligned}\|\boldsymbol{A}^t-\hat{\boldsymbol{A}}^t\|_2 &= \left\|\prod_{k=T-t}^{T-1}e^{\Delta[k]\boldsymbol{A}_c}-\prod_{k=T-t}^{T-1}e^{\hat{\Delta}[k]\hat{\boldsymbol{A}}_c}\right\|_2 \\ &= \left\|\sum_{i=T-t}^{T-1}\left(\left(\prod_{j=i}^{T-1}e^{\Delta[j]\boldsymbol{A}_c}\right)\left(\prod_{k=T-t}^{i-1}e^{\hat{\Delta}[k]\hat{\boldsymbol{A}}_c}\right)\right.\right. \\ &\qquad \left.\left.-\left(\prod_{j=i+1}^{T-1}e^{\Delta[j]\boldsymbol{A}_c}\right)\left(\prod_{k=T-t}^{i}e^{\hat{\Delta}[k]\hat{\boldsymbol{A}}_c}\right)\right)\right\|_2.\end{aligned} \tag{57}$$

Factor common terms to obtain

$$\begin{aligned}&\leq \left\|\sum_{i=T-t}^{T-1}\left(\prod_{j=i+1}^{T-1}e^{\Delta[j]\boldsymbol{A}_c}\right)\left(e^{\Delta[i]\boldsymbol{A}_c}-e^{\hat{\Delta}[i]\hat{\boldsymbol{A}}_c}\right)\left(\prod_{k=T-t}^{i-1}e^{\hat{\Delta}[k]\hat{\boldsymbol{A}}_c}\right)\right\|_2 \\ &\leq \sum_{i=T-t}^{T-1}\left\|\prod_{j=i+1}^{T-1}e^{\Delta[j]\boldsymbol{A}_c}\right\|_2\left\|e^{\Delta[i]\boldsymbol{A}_c}-e^{\hat{\Delta}[i]\hat{\boldsymbol{A}}_c}\right\|_2\left\|\prod_{k=T-t}^{i-1}e^{\hat{\Delta}[k]\hat{\boldsymbol{A}}_c}\right\|_2.\end{aligned} \tag{58}$$

Applying Lemma E.3, we get

$$\begin{aligned}&\leq \sum_{i=T-t}^{T-1}\rho_{\boldsymbol{A}}^{T-i-1}\|e^{\Delta[i]\boldsymbol{A}_c}-e^{\hat{\Delta}[i]\hat{\boldsymbol{A}}_c}\|_2\rho_{\boldsymbol{A}}^{i-T+t} \\ &= \sum_{i=T-t}^{T-1}\rho_{\boldsymbol{A}}^{t-1}\|e^{\Delta[i]\boldsymbol{A}_c}-e^{\hat{\Delta}[i]\hat{\boldsymbol{A}}_c}\|_2.\end{aligned} \tag{59}$$

Use Lemma C.3 to derive

$$\|\boldsymbol{A}^t-\hat{\boldsymbol{A}}^t\|_2 \leq \rho_{\boldsymbol{A}}^t\sum_{i=T-t}^{T-1}\|\Delta[i]\boldsymbol{A}_c-\hat{\Delta}[i]\hat{\boldsymbol{A}}_c\|_2. \tag{60}$$

Apply triangle inequality:

$$\begin{aligned}
&\leq \rho_{\boldsymbol{A}}^t \sum_{i=T-t}^{T-1} \left(\|\Delta[i](\boldsymbol{A}_c - \hat{\boldsymbol{A}}_c)\|_2 + \|(\Delta[i] - \hat{\Delta}[i])\hat{\boldsymbol{A}}_c\|_2 \right) \\
&\leq \rho_{\boldsymbol{A}}^t \sum_{i=T-t}^{T-1} \left(\mathfrak{M}_\Delta \epsilon_A + |\Delta[i] - \hat{\Delta}[i]| \mathfrak{B}_{\boldsymbol{A}} \right) \\
&\leq \rho_{\boldsymbol{A}}^t \sum_{i=T-t}^{T-1} \left(\mathfrak{M}_\Delta \epsilon_A + \epsilon_\Delta \mathfrak{B}_{\boldsymbol{A}} \right).
\end{aligned} \tag{61}$$

At last, we obtain the final bound:

$$\|\boldsymbol{A}^t - \hat{\boldsymbol{A}}^t\|_2 \leq t\rho_{\boldsymbol{A}}^t (\mathfrak{M}_\Delta \epsilon_A + \mathfrak{B}_{\boldsymbol{A}} \epsilon_\Delta). \tag{62}$$

□

Lemma E.6. *Let $\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}, \hat{\Delta}$ be covers for $\boldsymbol{A}, \boldsymbol{B}, \Delta$. Then,*

$$\begin{aligned}
&\left\| \sum_{t=0}^{T-1} \boldsymbol{A}^t \Delta[T-1-t] \boldsymbol{B}_c u[T-1-t] - \sum_{t=0}^{T-1} \hat{\boldsymbol{A}}^t \hat{\Delta}[T-1-t] \hat{\boldsymbol{B}}_c u[T-1-t] \right\|_2 \\
&\leq \mathfrak{M}_\Delta S_1 \epsilon_B + \mathfrak{M}_\Delta^2 \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 S_2 \epsilon_A + \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 \left(S_1 + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{A}} S_2 \right) \epsilon_\Delta
\end{aligned} \tag{63}$$

, where $S_1 = \frac{1-\rho_{\boldsymbol{A}}^T}{1-\rho_{\boldsymbol{A}}}$ and $S_2 = \frac{\rho_{\boldsymbol{A}}(1-\rho_{\boldsymbol{A}}^T)}{(1-\rho_{\boldsymbol{A}})^2} - \frac{T\rho_{\boldsymbol{A}}^T}{1-\rho_{\boldsymbol{A}}}$.

Proof. Write the LHS as follows:

$$\left\| \sum_{t=0}^{T-1} \left(\left(\boldsymbol{A}^t \Delta[T-1-t] \boldsymbol{B}_c - \hat{\boldsymbol{A}}^t \hat{\Delta}[T-1-t] \hat{\boldsymbol{B}}_c \right) u[T-1-t] \right) \right\|_2. \tag{64}$$

Add and subtract the terms $\sum_{t=0}^{T-1} \left(\boldsymbol{A}^t \Delta[T-1-t] \hat{\boldsymbol{B}}_c \right) u[t-t-1]$ and $\sum_{t=0}^{T-1} \left(\boldsymbol{A}^t \hat{\Delta}[T-1-t] \hat{\boldsymbol{B}}_c \right) u[t-t-1]$ to derive

$$\begin{aligned}
&\| \sum_{t=0}^{T-1} \left(\boldsymbol{A}^t \Delta[T-1-t] \boldsymbol{B}_c - \boldsymbol{A}^t \Delta[T-1-t] \hat{\boldsymbol{B}}_c \right) u[T-t-1] \\
&+ \sum_{t=0}^{T-1} \left(\boldsymbol{A}^t \Delta[T-1-t] \hat{\boldsymbol{B}}_c - \boldsymbol{A}^t \hat{\Delta}[T-1-t] \hat{\boldsymbol{B}}_c \right) u[T-t-1] \\
&+ \sum_{t=0}^{T-1} \left(\boldsymbol{A}^t \hat{\Delta}[T-1-t] \hat{\boldsymbol{B}}_c - \hat{\boldsymbol{A}}^t \hat{\Delta}[T-1-t] \hat{\boldsymbol{B}}_c \right) u[T-t-1] \|_2.
\end{aligned} \tag{65}$$

Apply the triangle inequality to get

$$\begin{aligned}
&\sum_{t=0}^{T-1} \Big(\|\boldsymbol{A}^t \Delta[T-1-t](\boldsymbol{B}_c - \hat{\boldsymbol{B}}_c) u[T-t-1]\|_2 \\
&\qquad + \|\boldsymbol{A}^t (\Delta[T-1-t] - \hat{\Delta}[T-1-t]) \hat{\boldsymbol{B}}_c u[T-t-1]\|_2 \\
&\qquad + \|(\boldsymbol{A}^t - \hat{\boldsymbol{A}}^t) \Delta[\hat{T-1-t}] \hat{\boldsymbol{B}}_c u[T-t-1]\|_2 \Big)
\end{aligned} \tag{66}$$

which is upper bounded by

$$\begin{aligned}
&\leq \sum_{t=0}^{T-1} \Big(\|\boldsymbol{A}^t\|_2 |\Delta[T-1-t]| \|(\boldsymbol{B}_c - \hat{\boldsymbol{B}}_c) u[T-t-1]\|_2 \\
&\qquad + \|\boldsymbol{A}^t\|_2 |\Delta[T-1-t] - \hat{\Delta}[T-1-t]| \|\hat{\boldsymbol{B}}_c u[T-t-1]\|_2 \\
&\qquad + \|(\boldsymbol{A}^t - \hat{\boldsymbol{A}}^t)\|_2 |\hat{\Delta}[T-1-t]| \|\hat{\boldsymbol{B}}_c u[T-t-1]\|_2 \Big).
\end{aligned} \tag{67}$$

The application of Lemmas E.3 and E.5 to bound $\|\boldsymbol{A}^t\|_2$ and cover $\|\boldsymbol{A}^t - \hat{\boldsymbol{A}}^t\|_2$ results in

$$\leq \sum_{t=0}^{T-1} \left(\rho_{\boldsymbol{A}}^t \mathfrak{M}_\Delta \epsilon_B + \rho_{\boldsymbol{A}}^t \epsilon_\Delta \|\hat{\boldsymbol{B}}_c\|_2 \mathfrak{B}_u + t\rho_{\boldsymbol{A}}^t (\mathfrak{M}_\Delta \epsilon_A + \mathfrak{B}_{\boldsymbol{A}} \epsilon_\Delta) \mathfrak{M}_\Delta \|\hat{\boldsymbol{B}}_c\|_2 \mathfrak{B}_u \right). \tag{68}$$

Apply Lemma C.1 to bound $\|\hat{\boldsymbol{B}}_c\|_2$:

$$\leq \sum_{t=0}^{T-1} \left(\rho_{\boldsymbol{A}}^t \mathfrak{M}_\Delta \epsilon_B + \rho_{\boldsymbol{A}}^t \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 \epsilon_\Delta + t\rho_{\boldsymbol{A}}^t \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 (\mathfrak{M}_\Delta \epsilon_A + \mathfrak{B}_{\boldsymbol{A}} \epsilon_\Delta) \right). \tag{69}$$

Breaking the summation into two parts leads to

$$\begin{aligned}
&\leq \left(\mathfrak{M}_\Delta \epsilon_B + \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 \epsilon_\Delta \right) \sum_{t=0}^{T-1} \rho_{\boldsymbol{A}}^t + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 (\mathfrak{M}_\Delta \epsilon_A + \mathfrak{B}_{\boldsymbol{A}} \epsilon_\Delta) \sum_{t=0}^{T-1} t\rho_{\boldsymbol{A}}^t \\
&\leq \left(\mathfrak{M}_\Delta \epsilon_B + \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 \epsilon_\Delta \right) \frac{1 - \rho_{\boldsymbol{A}}^T}{1 - \rho_{\boldsymbol{A}}} + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 (\mathfrak{M}_\Delta \epsilon_A + \mathfrak{B}_{\boldsymbol{A}} \epsilon_\Delta) \left(\frac{\rho_{\boldsymbol{A}}(1 - \rho_{\boldsymbol{A}}^T)}{(1 - \rho_{\boldsymbol{A}})^2} - \frac{T\rho_{\boldsymbol{A}}^T}{1 - \rho_{\boldsymbol{A}}} \right)
\end{aligned}$$

which completes the proof. □

Remark. Lemma E.3 is stated for sufficiently large t. In Lemmas E.5 and E.6 we still apply that bound when summing over all time indices. The step is legitimate because any sum $\sum_{t=0}^{T-1}(\cdot)$ can be split at an index t_0 for which the hypothesis of Lemma E.3 holds for $t \geq t_0$:

$$\sum_{t=0}^{T-1}(\cdot) = \sum_{t=0}^{t_0-1}(\cdot) + \sum_{t=t_0}^{T-1}(\cdot).$$

The first term involves only finitely many values of t and therefore contributes a constant that is absorbed into the leading $\mathcal{O}(\cdot)$ rate. The second (tail) term is where the bound of Lemma E.3 is used, and it determines the asymptotic dependence on T. Hence, omitting the constant part does not affect the final Big-$\mathcal{O}$ expression in the main theorem.

Lemma E.7. *Let $\hat{\boldsymbol{A}}_c, \hat{\boldsymbol{W}}_{\boldsymbol{B}}, \hat{\boldsymbol{W}}_{\boldsymbol{C}}, \hat{p}, \hat{q}$ be covers for $\boldsymbol{A}_c, \boldsymbol{W}_{\boldsymbol{B}}, \boldsymbol{W}_{\boldsymbol{C}}, p, q$. Then,*

$$\begin{aligned}
\left| w^\top y[T] - \hat{w}^\top \hat{y}[T] \right| \leq\ & \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^2 \mathfrak{M}_\Delta S_1 \epsilon_{W_B} + \mathfrak{M}_\Delta^2 \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3 S_2 \epsilon_A \\
&+ \left(\mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3 \right) (S_1 + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{A}} S_2) (\epsilon_p + \epsilon_q) \\
&+ \mathfrak{B}_w \mathfrak{B}_u \epsilon_{W_C} + \epsilon_w,
\end{aligned} \tag{70}$$

where S_1 and S_2 are defined as in Lemma E.6.

Proof. The proof follows from the sequential application of Lemmas C.5, C.6, and E.6, yielding:

$$\begin{aligned}
&\left| w^\top y[T] - \hat{w}^\top \hat{y}[T] \right| \\
&\leq \mathfrak{B}_w \left(\mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u \left(\mathfrak{M}_\Delta S_1 \epsilon_B + \mathfrak{M}_\Delta^2 \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 S_2 \epsilon_A + \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 \left(S_1 + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{A}} S_2 \right) \epsilon_\Delta \right) + \epsilon_C \right) + \epsilon_w.
\end{aligned} \tag{71}$$

Finally, we apply Lemmas C.7, C.8, and C.4 to relate the covers for $\boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\Delta}$ to the covers for $\boldsymbol{W}_{\boldsymbol{B}}, \boldsymbol{W}_{\boldsymbol{C}}, p, q$, completing the proof. □

Proof of Theorem 3.3. We aim to construct a cover for the space of all selective SSMs $\mathcal{F}_{\text{SSM}} = \{z[T] = w^\top y[T] : y[T] \text{ is described in (8)}\}$ which is parametrizes by $\Theta_{\text{SSM}} = \{\boldsymbol{A}_c, \boldsymbol{W}_{\boldsymbol{B}}, \boldsymbol{W}_{\boldsymbol{C}}, p, q, w\}$. Let's look at how much the output $w^\top y[T]$ changes if we move to the points in the ϵ-net constructing the cover. This is done in Lemma E.7. Thus, we need to choose $\epsilon_A, \epsilon_{W_B}, \epsilon_{w_C}, \epsilon_q, \epsilon_p$ and ϵ_w subject to the following:

$$\begin{aligned}
\epsilon = \ & \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^2 \mathfrak{M}_\Delta S_1 \epsilon_{W_B} + \mathfrak{M}_\Delta^2 \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3 S_2 \epsilon_A \\
&+ \left(\mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3 \right) (S_1 + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{A}} S_2) (\epsilon_p + \epsilon_q) \\
&+ \mathfrak{B}_w \mathfrak{B}_u \epsilon_{W_C} + \epsilon_w,
\end{aligned} \tag{72}$$

which relates the ϵ-cover of a selective SSM to corresponding covers for each parameter in Θ_{SSM} as in (70).

Choose the covering for $\boldsymbol{W}_B$ according to Lemma D.3 such that

$$\ln \mathcal{N}_\infty(\mathcal{F}_{W_B}, \epsilon_{W_B}, \|\cdot\|_2) \leq \frac{\mathfrak{B}_u^2 \mathfrak{M}_{\boldsymbol{B}}^2}{\epsilon_{W_B}} \ln(2Nd+1). \tag{73}$$

Similarly, choose the covering for $\boldsymbol{W}_C$ by replacing v in Lemma D.3 with $x[T]$ which is bounded as in Lemma E.4 to derive

$$\begin{aligned} \ln \mathcal{N}_\infty(\mathcal{F}_{W_C}, \epsilon_{W_C}, \|\cdot\|_2) &\leq \frac{\left(\mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^2 S_1\right)^2 \mathfrak{M}_{\boldsymbol{C}}^2}{\epsilon_{W_C}^2} \ln(2dNd+1) \\ &= \frac{\mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_u^4 \mathfrak{M}_\Delta^2 \mathfrak{M}_{\boldsymbol{C}}^2 S_1^2}{\epsilon_{W_C}^2} \ln(2Nd^2+1). \end{aligned} \tag{74}$$

Likewise, choose the cover for w such that

$$\begin{aligned} &\ln \mathcal{N}_\infty(\mathcal{F}_w, \epsilon_w, \|\cdot\|_2) \leq \frac{\left(\mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_u^3 S_1\right)^2 \mathfrak{M}_w^2}{\epsilon_w^2} \ln(2d+1) \\ &= \frac{\mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_{\boldsymbol{C}}^2 \mathfrak{B}_u^6 \mathfrak{M}_\Delta^2 \mathfrak{M}_w^2 S_1^2}{\epsilon_w^2} \ln(2d+1). \end{aligned} \tag{75}$$

Lemma D.2 gives us the upper bound on the covering number for $\boldsymbol{A}_c$:

$$\ln \mathcal{N}(\mathcal{F}_{A_c}, \epsilon_A, \|\cdot\|_2) \leq \frac{2\mathfrak{M}_A^2 Nd}{\epsilon_A^2} \ln(\sqrt{2}Nd). \tag{76}$$

We may use Lemma D.3 again to cover q:

$$\ln \mathcal{N}_\infty(\mathcal{F}_q, \epsilon_q, \|\cdot\|_2) \leq \frac{\mathfrak{B}_u^2 \mathfrak{M}_q^2}{\epsilon_q^2} \ln(2d+1) \tag{77}$$

and p is covered simply by

$$\mathcal{N}_\infty(\mathcal{F}_p, \epsilon_p, \|\cdot\|_2) \leq \frac{2|p|}{\epsilon_p}. \tag{78}$$

Ignore the logarithmic dependencies and assume $\mathfrak{M}_{\boldsymbol{C}} = \mathfrak{B}_{\boldsymbol{C}}, \mathfrak{M}_{\boldsymbol{B}} = \mathfrak{B}_{\boldsymbol{B}}, \mathfrak{M}_w = \mathfrak{B}_w, \mathfrak{M}_q = \mathfrak{B}_q, \mathfrak{M}_{\boldsymbol{A}} = \mathfrak{B}_{\boldsymbol{A}}$ for simplicity. Construct the cover for the space of all selective SSMs $\mathcal{F}_{\text{SSM}}$ as the Cartesian product of all covers for each parameter in Θ_{SSM}. Then, the log covering number would be the sum of the log covering numbers of all parameters. Use Lemma C.9 to find $\epsilon_A, \epsilon_{W_B}, \epsilon_{w_C}, \epsilon_q, \epsilon_q$, and ϵ_w such that the size of total cover would be minimum:

$$\begin{aligned} &\epsilon^2 \ln \mathcal{N}_\infty(\mathcal{F}_{\text{SSM}}, \epsilon, \|\cdot\|_2) \\ &\leq \tilde{\mathcal{O}}\Big(\Big((\mathfrak{B}_u^2 \mathfrak{B}_{\boldsymbol{B}}^2)^{1/3} (\mathfrak{B}_w \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^2 \mathfrak{M}_\Delta S_1)^{2/3} + (\mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_u^4 \mathfrak{M}_\Delta^2 \mathfrak{B}_{\boldsymbol{C}}^2 S_1^2)^{1/3} (\mathfrak{B}_w \mathfrak{B}_u)^{2/3} \\ &+ (\mathfrak{B}_{\boldsymbol{A}}^2 Nd)^{1/3} (\mathfrak{M}_\Delta^2 \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3 S_2)^{2/3} + (\mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_{\boldsymbol{C}}^2 \mathfrak{B}_u^6 \mathfrak{M}_\Delta^2 \mathfrak{B}_w^2 S_1^2)^{1/3} \\ &+ (\mathfrak{B}_u^2 \mathfrak{B}_q^2)^{1/3} \left(\mathfrak{B}_w \mathfrak{B}_{\boldsymbol{B}} \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^3\right)^{2/3} \left(S_1 + \mathfrak{M}_\Delta \mathfrak{B}_{\boldsymbol{A}} S_2\right)^{2/3}\Big)^3\Big). \end{aligned} \tag{79}$$

in which we ignored the cover for p as it is dominated by other terms.

$$\begin{aligned} &\leq \tilde{\mathcal{O}}\Big(\Big(\mathfrak{M}_\Delta^{2/3} \mathfrak{B}_w^{2/3} \mathfrak{B}_u^2 \mathfrak{B}_{\boldsymbol{B}}^{2/3} \mathfrak{B}_{\boldsymbol{C}}^{2/3} S_1^{2/3} + \mathfrak{M}_\Delta^{2/3} \mathfrak{B}_w^{2/3} \mathfrak{B}_u^2 \mathfrak{B}_{\boldsymbol{B}}^{2/3} \mathfrak{B}_{\boldsymbol{C}}^{2/3} S_1^{2/3} \\ &+ \mathfrak{M}_\Delta^{4/3} \mathfrak{B}_{\boldsymbol{A}}^{2/3} \mathfrak{B}_w^{2/3} \mathfrak{B}_u^2 \mathfrak{B}_{\boldsymbol{B}}^{2/3} \mathfrak{B}_{\boldsymbol{C}}^{2/3} S_2^{2/3} N^{1/3} d^{1/3} + \mathfrak{M}_\Delta^{2/3} \mathfrak{B}_w^{2/3} \mathfrak{B}_u^2 \mathfrak{B}_{\boldsymbol{B}}^{2/3} \mathfrak{B}_{\boldsymbol{C}}^{2/3} S_1^{2/3} \\ &+ \mathfrak{M}_\Delta^{2/3} \mathfrak{B}_{\boldsymbol{A}}^{2/3} \mathfrak{B}_q^{2/3} \mathfrak{B}_w^{2/3} \mathfrak{B}_u^{8/3} \mathfrak{B}_{\boldsymbol{B}}^{2/3} \mathfrak{B}_{\boldsymbol{C}}^{2/3} S_2^{2/3}\Big)^3\Big), \end{aligned} \tag{80}$$

where we used the fact that S_1 is dominated by S_2 for large T to obtain the last term. Therefore, we have

$$\begin{aligned} &\leq \tilde{\mathcal{O}}\Big(\mathfrak{M}_\Delta^2 \mathfrak{B}_w^2 \mathfrak{B}_u^6 \mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_{\boldsymbol{C}}^2 \Big(S_1^{2/3} + S_1^{2/3} + \mathfrak{M}_\Delta^{2/3} \mathfrak{B}_{\boldsymbol{A}}^{2/3} S_2^{2/3} N^{1/3} d^{1/3} \\ &\quad + 1 + \mathfrak{B}_{\boldsymbol{A}}^{2/3} \mathfrak{B}_q^{2/3} \mathfrak{B}_u^{2/3} S_2^{2/3}\Big)^3\Big). \end{aligned} \tag{81}$$

Ignoring the constant terms and S_1 compared to S_2 results in

$$\leq \tilde{\mathcal{O}}\left(\mathfrak{M}_\Delta^2 \mathfrak{B}_w^2 \mathfrak{B}_u^6 \mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_{\boldsymbol{C}}^2 \mathfrak{B}_{\boldsymbol{A}}^2 S_2^2 (\mathfrak{M}_\Delta^{2/3} N^{1/3} d^{1/3} + \mathfrak{B}_q^{2/3} \mathfrak{B}_u^{2/3})^3\right). \tag{82}$$

The square root of this expression is $\mathcal{C}_{\mathcal{F}}$. The proof is complete by the application of Lemma D.4. □

F Proof for Proposition 3.4: Genralization Error Bound for Linear Attentions

Proof of Proposition 3.4. The proof that follows is similar to the proof of Theorem 3.3 with modifications accounting for the simplifications made in Assumptions 3.5 and 3.6. Specifically, we do not need to cover $\boldsymbol{A}_c$, p, or q. Therefore, Lemma E.6 simplifies to

$$\Big\| \sum_{t=0}^{T-1} (\boldsymbol{W}_B - \widehat{\boldsymbol{W}}_B)\, u[T-1-t] \Big\|_2 \;\leq\; T\, \epsilon_{W_B}. \tag{83}$$

Consequently, Lemma E.7 becomes

$$\left| w^\top y[T] - \widehat{w}^\top \widehat{y}[T] \right| \;\leq\; T\, \mathfrak{B}_w\, \mathfrak{B}_{\boldsymbol{C}}\, \mathfrak{B}_u^2\, \epsilon_{W_B} \;+\; \mathfrak{B}_w\, \mathfrak{B}_u\, \epsilon_{W_C} \;+\; \epsilon_w. \tag{84}$$

Also, Lemma E.4 reduces to

$$\|x[T]\|_2 \;\leq\; T\, \mathfrak{B}_{\boldsymbol{B}}\, \mathfrak{B}_u^2, \tag{85}$$

which is used to cover $\boldsymbol{W}_C$. Hence (79) becomes

$$\begin{aligned} \epsilon^2 \ln \mathcal{N}_\infty(\mathcal{F}_{LA}, \epsilon, \|\cdot\|_2) \leq \widetilde{\mathcal{O}}\Big(& (\mathfrak{B}_u^2 \mathfrak{B}_{\boldsymbol{B}}^2)^{\frac{1}{3}} (T\, \mathfrak{B}_w \mathfrak{B}_{\boldsymbol{C}} \mathfrak{B}_u^2)^{\frac{2}{3}} \\ & + (T^2\, \mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_u^4 \mathfrak{B}_{\boldsymbol{C}}^2)^{\frac{1}{3}} (\mathfrak{B}_w \mathfrak{B}_u)^{\frac{2}{3}} + (T^2\, \mathfrak{B}_{\boldsymbol{B}}^2 \mathfrak{B}_{\boldsymbol{C}}^2 \mathfrak{B}_u^6 \mathfrak{B}_w^2)^{\frac{1}{3}} \Big)^3 \\ = \widetilde{\mathcal{O}}(& T^2\, \mathfrak{B}_{\boldsymbol{B}}^2\, \mathfrak{B}_{\boldsymbol{C}}^2\, \mathfrak{B}_u^6\, \mathfrak{B}_w^2). \end{aligned} \tag{86}$$

By applying Lemma D.4, the proof is complete. □

G Proof for Theorem 4.1: Lower Bound on the Rademacher Complexity

Proof of Theorem 4.1. We consider a restricted class of scalar ($d = N = 1$) selective SSMs defined by

$$\Theta_l = \left\{ \boldsymbol{A}_c = \ln(1 + s_{\boldsymbol{A}}),\; \boldsymbol{W}_B = 1,\; \boldsymbol{W}_C = 1,\; p = e,\; q = 0,\; w \;\middle|\; |w| \leq \mathfrak{B}_w \right\}. \tag{87}$$

Since $\Theta_l \subset \Theta_{\text{SSM}}$, the Rademacher complexity of this restricted class provides a lower bound for that of the full class $\mathcal{F}_{\text{SSM}}$. For this class the step size would be fixed $\Delta[k] = 1$, and the discretized matrices become

$$\boldsymbol{A}[k] = 1 + s_{\boldsymbol{A}}, \quad \boldsymbol{B}[k] = u[k], \quad \boldsymbol{C}[k] = u[k]. \tag{88}$$

Thus, the resulting state space recurrence is

$$x[k] = (1 + s_{\boldsymbol{A}}) x[k-1] + u[k]^2, \qquad y[k] = u[k] x[k], \qquad z[k] = w y[k]. \tag{89}$$

With constant input $u[k] = 1$, the closed-form expression for the output becomes

$$z[T] = w \frac{(1 + s_{\boldsymbol{A}})^T - 1}{s_{\boldsymbol{A}}}. \tag{90}$$

Hence, the hypothesis class can be expressed as

$$\mathcal{F}_l = \left\{ w \frac{(1 + s_{\boldsymbol{A}})^T - 1}{s_{\boldsymbol{A}}} \;\middle|\; |w| \leq \mathfrak{B}_w \right\}. \tag{91}$$

The empirical Rademacher complexity is then

$$\text{Rad}_{\mathcal{S}}(\mathcal{F}_l) = \frac{1}{m} \mathbb{E}_\sigma \left[\sup_{|w| \leq \mathfrak{B}_w} \sum_{i=1}^m \sigma_i z_{(i)}[T] \right] = \frac{(1 + s_{\boldsymbol{A}})^T - 1}{s_{\boldsymbol{A}}} \frac{1}{m} \mathbb{E}_\sigma \left[\sup_{|w| \leq \mathfrak{B}_w} w \sum_{i=1}^m \sigma_i \right]. \tag{92}$$

The supremum is achieved when $w = \mathfrak{B}_w \operatorname{sign}\left(\sum_{i=1}^m \sigma_i\right)$, yielding

$$\operatorname{Rad}_{\mathcal{S}}(\mathcal{F}_l) = \mathfrak{B}_w \frac{(1+s_{\boldsymbol{A}})^T - 1}{s_{\boldsymbol{A}}} \sqrt{\frac{2}{\pi m}}. \tag{93}$$

When $s_{\boldsymbol{A}} = 0$, the recursion becomes $x[k] = x[k-1] + 1$, so $x[T] = T$, and hence $z[T] = wT$. The resulting hypothesis class is $\{wT \mid |w| \leq \mathfrak{B}_w\}$, and a similar argument yields

$$\operatorname{Rad}_{\mathcal{S}}(\mathcal{F}_l) = \mathfrak{B}_w T \sqrt{\frac{2}{\pi m}}. \tag{94}$$

□

H Extra Discussion

Note that the term S_2 in Theorem 3.3, derived in Lemma D.6) is

$$S_2 = \frac{\rho_A\left(1-\rho_A^T\right)}{(1-\rho_A)^2} - \frac{T\,\rho_A^T}{1-\rho_A}.$$

This expression does not diverge as $\rho_A \to 1$ even though there is a $1-\rho_A$ in the denominator. Applying L'Hôpital's rule twice yields

$$S_2 = \frac{T^2 - T}{2},$$

which is finite. Therefore, although increasing ρ_A degrades generalization, the generalization bound does not blow up at $\rho_A = 1$; rather, it remains bounded and grows quadratically with T. The generalization becomes severely damaged only when $\rho_A > 1$, where the bound becomes exponentially increasing in $\mathcal{O}(T\rho_A^T)$.

Proof that the bound remains finite as $\rho \to 1$. We compute

$$\lim_{\rho\to 1}\left(\frac{\rho(1-\rho^T)}{(1-\rho)^2} - \frac{T\rho^T}{1-\rho}\right) = \lim_{\rho\to 1}\frac{\rho - T\rho^T + (T-1)\rho^{T+1}}{(1-\rho)^2}.$$

First derivatives:

$$\frac{d}{d\rho}\big(\text{numerator}\big) = 1 - T^2\rho^{T-1} + (T^2-1)\rho^T, \qquad \frac{d}{d\rho}\big(\text{denominator}\big) = -2(1-\rho).$$

Second derivatives:

$$\frac{d^2}{d\rho^2}\big(\text{numerator}\big) = -T^2(T-1)\rho^{T-2} + T(T^2-1)\rho^{T-1}, \qquad \frac{d^2}{d\rho^2}\big(\text{denominator}\big) = 2.$$

Evaluating at $\rho \to 1$ gives

$$\lim_{\rho\to 1}\left(\frac{\rho(1-\rho^T)}{(1-\rho)^2} - \frac{T\rho^T}{1-\rho}\right) = \frac{T^2-T}{2}.$$

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 contributions are summarized at the end of Section 1 under three categories: Theoretical, Analytical, and Empirical. They are also explained in detail in the preceding paragraph and in the abstract.

   Guidelines:

   - The answer NA 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 NA 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: We discussed the limitations in a dedicated section.

   Guidelines:

   - The answer NA 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: We present the assumptions for the main theorem in Assumptions 3.1–3.4, and justify them immediately following their statement. A proof sketch is included in the main text, while the full proof, with all technical details and supporting lemmas, is provided in the Appendix. Assumptions for other theoretical results are also stated precisely, and all corresponding proofs are given rigorously in the Appendix.

Guidelines:

- The answer NA 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: Our code is accessible at: `https://github.com/Arya-Honarpisheh/gen_err_sel_ssm`. The model used in the experiments is implemented exactly as described in Section 2.2. Training follows standard procedures, and all necessary information for reproducibility is provided in the paper. Additional implementation details are included in Section B, as they could not fit within the main paper due to space constraints.

Guidelines:

- The answer NA 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: [Yes]

   Justification: Our code is accessible at: https://github.com/Arya-Honarpisheh/gen_err_sel_ssm.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.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://nips.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, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provided all the relevant details of the experiments. Note that additional implementation details are provided in Section B, as they did not fit within the page limit of the main paper.

   Guidelines:

   - The answer NA 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: The main experiment of this paper is repeated five times. The corresponding figures include error bars to illustrate the statistical significance of the theoretical claims.

Guidelines:

- The answer NA 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: [No]

   Justification: Our experiments were conducted solely to evaluate the generalization performance of the model, without considering any metrics related to computational efficiency. Therefore, we did not report compute resources, as they are irrelevant in the absence of compute time or related performance indicators, which fall outside the scope of our interest.

   Guidelines:

   - The answer NA 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 presented in this paper is fully consistent with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA 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: [NA]

Justification: This is a theoretical paper focused on understanding deep learning models, and it poses no societal impact concerns.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This is a theoretical study focused on understanding the fundamental properties of deep learning models. As such, it does not involve any high risk or misuse.

Guidelines:

- The answer NA 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: We implemented our models and experiments in Python using PyTorch. We used the IMDb dataset [38] and ListOps Nangia and Bowman [39], which are publicly available and were used under their standard terms of use. Additionally, we used a small code snippet from a Substack article by Keller [37], which is publicly available. The author retains ownership of the content, and no explicit license is provided. We have credited the author in our paper and provided a link to the source. The code was used solely for research purposes and was not redistributed or modified beyond what is permitted under fair use.

Guidelines:

- The answer NA 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: [Yes]

Justification: We have provided the code to reproduce the experiments of this paper. It comes with the same license as the paper: CC BY-NC-SA 4.0.

Guidelines:

- The answer NA 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: [NA]

Justification:

Guidelines:

- The answer NA 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: [NA]

    Justification:

    Guidelines:

    - The answer NA 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 rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA 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 (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.