State Space Models are Comparable to Transformers in Estimating Functions with Dynamic Smoothness

Naoki Nishikawa [1] **Taiji Suzuki** [1 2]

Abstract

While the capabilities of deep neural networks based on state space models (SSMs) have been primarily investigated through experimental comparisons, theoretical understanding is still limited. In particular, there is a lack of statistical and quantitative evaluation of whether SSMs can replace Transformers. In this paper, we theoretically explore in which tasks SSMs can be alternatives to Transformers from the perspective of estimating sequence-to-sequence functions. We consider the setting where the target function has direction-dependent smoothness, and prove that SSMs can estimate such functions with the same convergence rate as Transformers. Additionally, we prove that SSMs can estimate the target function as effectively as Transformers, even if the smoothness changes depending on the input sequence. Our results suggest that SSMs can replace Transformers when estimating the functions in certain classes that appear in practice.

1. Introduction

Foundation models based on Transformers have achieved remarkable success in various sequence modeling tasks such as natural language processing (Vaswani et al., 2017), computer vision (Dosovitskiy et al., 2020), and speech recognition (Radford et al., 2023). The superior performance of Transformers is attributed to the self-attention mechanism, which enables the model to aggregate the information from the input sequence.

In contrast to its success, the self-attention mechanism has a potential problem that it requires a large amount of computation and memory. To deal with this issue, many studies have attempted to develop efficient models that can replace Transformers. Among them, *Structured State Space Models* (SSMs) have garnered considerable interest recently. One advantage of SSMs is that the output can be computed with a significantly small time using convolution via FFT algorithm or recursive computation. Based on the original SSMs, many improvements have been proposed, such as HiPPO-based intialization (Gu et al., 2021) and architectures using gated convolutions (Fu et al., 2022; Poli et al., 2023).

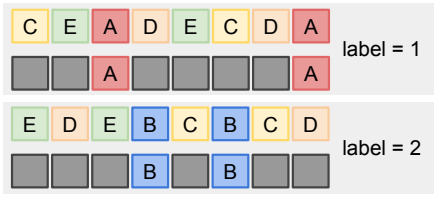

Figure 1.1. Conceptual illustrations of the abilities of SSMs. A classification task in which the existence of the token "A" or "B" is important. We prove that SSMs can extract such tokens, even if the position of the important token is *different for each input*.

Networks based on SSMs have accomplished high performance in various applications such as gene analysis (Nguyen et al., 2024), audio generation (Goel et al., 2022) and speech recognition (Saon et al., 2023). On the other hand, some of the recent studies such as Merrill et al. (2024) pointed out the limitations of SSMs, especially for their abilities to solve tasks. Therefore, it is still unclear in what situation we can replace Transformers with SSMs.

Recently, some studies have theoretically investigated the abilities of SSMs. For instance, Wang and Xue (2024) show that SSMs are universal approximators for continuous sequence-to-sequence functions. Additionally, Massaroli et al. (2024) constructed the parameters of SSMs to solve the task called associated recall. However, they mainly focus on the expressive power of SSMs and do not provide statistical understanding. Furthermore, quantitative evaluations to compare SSMs and Transformers are limited.

Our contributions. In this paper, we explore the abilities of *SSMs with gated convolution* to replace Transformers from the perspective of statistical learning theory. More specifically, we investigate the estimation ability of SSMs for the function classes called γ-*smooth* and *piecewise* γ-*smooth*. For the function in these classes, Takakura and Suzuki (2023) showed that Transformers can estimate them effectively. We prove that SSMs can also estimate those

[1]The University of Tokyo, Tokyo, Japan [2]RIKEN AIP, Tokyo, Japan. Correspondence to: Naoki Nishikawa <nishikawa-naoki259@g.ecc.u-tokyo.ac.jp>.

Work presented at TF2M workshop at ICML 2024, Vienna, Austria. PMLR 235, 2024.

functions with the same convergence rate as Transformers, and show that SSMs can replace Transformers when estimating those functions.

The essential point of the two function classes above is that they have smoothness structures. As for γ-smooth functions, the smoothness of the function is the same for all input sequences, i.e., the important features to extract are fixed. On the other hand, piecewise γ-smooth functions have different smoothness depending on the input. This function class characterizes the ability of Transformers and SSMs to extract important features *dynamically*.

Notations. For a set $S \subseteq \mathbb{R}$ and $d \in \mathbb{N}$, let $S^{d\times\infty} := \left\{[\dots, s_{-2}, s_{-1}, s_0] \mid s_i \in S^d\right\}$. For $F : \Omega \to \mathbb{R}^l$, let $\|F\|_\infty := \sup_{X\in\Omega} \|F(X)\|_\infty$. For the probability measure P_X on Ω and $p > 0$, the norm $\|\cdot\|_{p,P_X}$ is defined by $\|f\|_{p,P_X} = \left(\int_\Omega \|f(X)\|_p^p \,\mathrm{d}P_X\right)^{1/p}$. For a matrix A, let $\|A\|_0 = |\{(i,j) \mid A_{ij} \neq 0\}|$. For $F\colon \mathbb{R}^{d\times\infty} \to \mathbb{R}^{1\times\infty}$, we denote $F = (F_j)_{j=-\infty}^{\infty}$ with $F_j\colon [0,1]^{d\times\infty} \to \mathbb{R}$.

2. Problem Settings

In this paper, we consider a non-parametric regression problem where the input is infinite-dimensional. More concretely, we suppose that the input $X := [x_i]_{i=-\infty}^{0} \in \mathbb{R}^{d\times\infty}$ is a sequence of d-dimensional tokens. Let P_X be a probability measure on $([0,1]^{d\times\infty}, \mathcal{B}([0,1]^{d\times\infty}))$, and denote $\Omega := \mathrm{supp}(P_X)$. We assume that P_X is shift-invariant, i.e., for any $i \in \mathbb{Z}$ and $B \in \mathcal{B}([0,1]^{d\times\infty})$, it holds that $P_X(B) = P_X(\{\Sigma_i(X) \mid X \in B\})$ for any $i \in \mathbb{N}$, where Σ_j is the shift operator $\Sigma_j\colon \mathbb{R}^{d\times\infty} \to \mathbb{R}^{d\times\infty}$ defined by $(\Sigma_j(X))_i = x_{i-j}$ for $X = [x_j]_{j=-\infty}^{0} \in \mathbb{R}^{d\times\infty}$.

We also consider infinite-dimensional outputs. As same as the usual nonparametric regression setting, suppose that we observe n i.i.d. inputs $X^{(i)} \sim P_X$ $(i = 1, \dots, n)$ and the corresponding outputs $Y^{(i)} \in \mathbb{R}^{1\times\infty}$ generated by $Y^{(i)} = F^\circ(X^{(i)}) + \xi^{(i)}$, where $\xi^{(i)} \in \mathbb{R}^{d\times\infty}$ is the i.i.d. noise generated from $\mathcal{N}(0, \sigma^2)$ $(\sigma > 0)$. We further assume that $\left\{\xi^{(i)}\right\}_{i=1}^n$ is independent of the inputs $\left\{X^{(i)}\right\}_{i=1}^n$.
Given the pairs of inputs and outputs $\left\{(X^{(i)}, Y^{(i)})\right\}_{i=1}^n$, we obtain the estimator $\hat{F}$ of the target function F through empirical risk minimization:

$$\hat{F} = \arg\min_{F\in\mathcal{S}} \frac{1}{n}\sum_{i=1}^{n}\sum_{j=l}^{r}\left(Y^{(i)} - F^\circ(X^{(i)})\right)^2,$$

where $\mathcal{S}$ is supposed to be the class of networks that we define in the next section. To measure the statistical performance of the estimator $\hat{F}$, we utilize mean squared error (MSE) defined by

$$R_{l,r}(\hat{F}, F^\circ) = \frac{1}{l-r+1}\sum_{j=l}^{r}\mathbb{E}\left[\left\|\hat{F}_j(X) - F_j^\circ(X)\right\|_{2,P_X}^2\right],$$

where the expectation is taken for $\left\{\xi^{(i)}\right\}_{i=1}^n$. Note that we consider the average error over the finite segment $[l : r]$ to avoid the technical difficulty of infinite-dimensionality. However, the estimation error analysis in Section 4 is independent of the choice of l and r.

Similar to Takakura and Suzuki (2023), we assume that the target function is *shift equivariant*. That is, the target function $F\colon \mathbb{R}^{d\times\infty} \to \mathbb{R}^{d\times\infty}$ satisfies $F(\Sigma_j(X)) = \Sigma_j(F(X))$ for any $j \in \mathbb{N}$. This is a natural assumption in various applications such as language processing, audio processing, and time-series analysis.

2.1. γ-smooth function class

Here, we introduce the γ-smooth function class, which was first proposed by Okumoto and Suzuki (2021). First of all, for $r \in \mathbb{Z}_0^{d\times\infty}$, we define $\psi_{r_{ij}} : [0,1] \to \mathbb{R}$ by

$$\psi_{r_{ij}}(x) := \begin{cases} \sqrt{2}\cos(2\pi|r_{ij}|x) & (r_{ij} < 0), \\ 1 & (r_{ij} = 0), \\ \sqrt{2}\sin(2\pi|r_{ij}|x) & (r_{ij} > 0), \end{cases}$$

and $\psi_r : [0,1]^{d\times\infty} \to \mathbb{R}$ by $\psi_r(X) = \prod_{i=1}\prod_{j=1}\psi_{r_{ij}}(X_{ij})$. Then, $\{\psi_r\}_{r\in\mathbb{Z}_0^{d\times\infty}}$ forms a complete orthonormal system of $L^2([0,1]^{d\times\infty})$, Therefore, any $f \in L^2([0,1]^{d\times\infty})$ can be expanded as $f = \sum_{r\in\mathbb{Z}_0^{d\times\infty}}\langle f, \psi_r\rangle\psi_r$. For $s \in \mathbb{N}_0^{d\times\infty}$, we define $\delta_s(f) := \sum_{r\in\mathbb{Z}_0^{d\times\infty}, \lfloor 2^{s_{ij}-1}\rfloor \le r_{ij} < 2^{s_{ij}}}\langle f, \psi_r\rangle\psi_r$. Then, we define the γ-smooth function class as follows.

Definition 2.1 (γ-smooth function class)**.** For a given $\gamma : \mathbb{N}_0^{d\times\infty} \to \mathbb{R}$ which is monotonically non-decreasing with respect to each coordinate and $p \ge 2, \theta \ge 1$, we define the γ-smooth function space as follows:

$$\mathcal{F}_{p,\theta}^{\gamma}([0,1]^{d\times\infty}) := \left\{f \in L^2([0,1]^{d\times\infty}) \mid \|f\|_{\mathcal{F}_{p,\theta}^{\gamma}} < \infty\right\},$$

where $\|f\|_{\mathcal{F}_{p,\theta}^{\gamma}} := \left(\sum_{s\in\mathbb{N}_0^{d\times\infty}} 2^{\theta\gamma(s)}\|\delta_s(f)\|_{p,P_X}^\theta\right)^{1/\theta}$. We also define the finite dimensional version of γ-smooth function space $\mathcal{F}_{p,\theta}^{\gamma}([0,1]^{d\times l})$ for $l \in \mathbb{N}$ in the same way.

Note that $\delta_s(f)$ can be seen as the frequency component of f with frequency $|r_{ij}| \sim 2^{s_{ij}}$ toward each coordinate. Therefore, we can interpret that γ controls the amplitude of each frequency component through weighting the term $\|\delta_s(f)\|_{p,P_X}$ in the norm. In other words, if $\gamma(s)$ is larger, the norm of frequency component $\delta_s(f)$ is smaller.

As a special case of γ, we consider the following two types of smoothness:

Definition 2.2 (Mixed and anisotropic smoothness)**.** Let $a \in \mathbb{R}_{>0}^{d\times\infty}$ be a smoothness parameter. Then, *mixed smoothness* is defined by $\gamma(s) = \langle a, s\rangle$. Additionally, *anisotropic smoothness* is defined by $\gamma(s) = \max\{a_{ij}s_{ij} \mid i \in [d], j \in \mathbb{Z}\}$.

For each $i \in [d], j \in \mathbb{Z}$, the parameter a_{ij} can be viewed as the smoothness for the coordinate X_{ij}. When a_{ij} is large, $\gamma(s)$ with $s_{ij} \neq 0$ increases and the frequency component $\delta_s(f)$ with $s_{ij} \neq 0$ becomes smaller. In contrast, small a_{ij} implies that the function is not smooth towards the coordinate (i, j), which implies that X_{ij} is an *important feature*.

The function class $\mathcal{F}^{\gamma}_{p,\theta}([0,1]^{d\times\infty})$ can be seen as extension of some famous function spaces to the infinite-dimensional setting. Indeed, if P_X is uniform distribution on $[0,1]^{1\times l}$ and $p < \infty$, then $\mathcal{F}^{\gamma}_{p,\theta}([0,1]^{1\times l})$ with mixed smoothness is equivalent to the mixed-Besov space. Moreover, if P_X is uniform distribution, then the anisotropic Sobolev space included in the unit ball of $\mathcal{F}^{\gamma}_{2,2}$ with anisotropic smoothness.

We also introduce some notation related to the smoothness parameter a. Let $\bar{a}$ be the sequence obtained by sorting a in ascending order. That is, $\bar{a} = [a_{i_1,j_1}, \dots, a_{i_k,j_k}, \dots]$ satisfies $a_{i_k,j_k} \leq a_{i_{k+1},j_{k+1}}$ for any $k \in \mathbb{N}$. Then, we define weak l^{α}-norm for $\alpha > 0$ as $\|a\|_{wl^{\alpha}} := \sup_j j^{\alpha} \bar{a}_j^{-1}$. Additionally, we define $a^{\dagger} = \bar{a}_1$ for the mixed smoothness and $a^{\dagger} = \left(\sum_{i=1}^{\infty} \bar{a}_i^{-1}\right)^{-1}$ for the anisotropic smoothness.

2.2. Piecewise γ-smooth function class

We now describe the piecewise γ-smooth function class. This was proposed by Takakura and Suzuki (2023) to clarify the advantage of Transformers compared to CNNs. More specifically, through the estimation abilities for the function class, they showed that Transformers have ability to determine which features to extract depending on the input. The rigourous definition is as follows.

Definition 2.3 (Piecewise γ-smooth function class)**.** For an index set Λ, let $\{\Omega_\lambda\}_{\lambda\in\Lambda}$ be a disjoint partition of Ω. That is, $\{\Omega_\lambda\}_{\lambda\in\Lambda}$ satisfies $\Omega = \bigcup_{\lambda\in\Lambda} \Omega_\lambda$, $\Omega_\lambda \cap \Omega_{\lambda'} = \emptyset$ $(\lambda \neq \lambda')$. For $V \in \mathbb{N}$ and a set of bijections $\{\pi_\lambda\}_{\lambda\in\Lambda}$ between $[V+1]$ and $[-V:0]$, define $\Pi_\lambda : \mathbb{R}^{d\times[-V:0]} \to \mathbb{R}^{d\times(V+1)}$ and $\Pi : \Omega \to \mathbb{R}^{d\times(V+1)}$ by

$$\Pi_\lambda([x_{-V}, \dots, x_0]) := [x_{\pi_\lambda(1)}, \dots, x_{\pi_\lambda(V+1)}],$$
$$\Pi(X) := \Pi_\lambda(X[-V:0]) \quad \text{if } X \in \Omega_\lambda.$$

Then, for $p \geq 2, \theta \geq 1$ and $\gamma : \mathbb{N}_0^{d\times\infty} \to \mathbb{R}$, we define piecewise γ-smooth function class $\mathcal{P}^{\gamma}_{p,\theta}(\Omega)$ as the set of functions $g := f \circ \Pi$ that satisfying $f \in \mathcal{F}^{\gamma}_{p,\theta}([0,1]^{d\times(V+1)})$ and $\|g\|_{\mathcal{P}^{\gamma}_{p,\theta}} < \infty$, where the norm $\|g\|_{\mathcal{P}^{\gamma}_{p,\theta}}$ is defined by $\|g\|_{\mathcal{P}^{\gamma}_{p,\theta}} := \left(\sum_{s\in\mathbb{N}_0^{d\times[-V:0]}} 2^{\theta\gamma(s)} \|\delta_s(f) \circ \Pi\|^{\theta}_{p,P_X}\right)^{1/\theta}$.

For $g \in \mathcal{P}^{\gamma}_{p,\theta}(\Omega)$ and $X \in \Omega_{\lambda_1}, Y \in \Omega_{\lambda_2}$ with $\lambda_1 \neq \lambda_2$, the smoothness parameter of g at X and Y are different. This means that the coordinates of important features can change depending on the input. Takakura and Suzuki (2023) gives the convergence rate of the estimation error, and show that it almost coincides the rate for γ-smooth functions.

To express how the disjoint partitions $\{\Omega_\lambda\}_{\lambda\in\Lambda}$ are determined, the *importance function* is defined as follows.

Definition 2.4 (importance function)**.** A function $\mu : \Omega \to \mathbb{R}^{\infty}$ is called an *importance function* for $\{\Omega_\lambda\}_{\lambda\in\Lambda}$ if μ satisfies $\Omega_\lambda = \{X \in \Omega \mid \mu(X)_{\pi_\lambda(1)} > \cdots > \mu(X)_{\pi_\lambda(V+1)}\}$.

In simple terms, the paritions $\{\Omega_\lambda\}_{\lambda\in\Lambda}$ are determined to sort the inputs in descending order of the importance function. As same as Takakura and Suzuki (2023), we assume that an importance function μ is *well-separated*, i.e., for some constant $c, \beta > 0$, μ satisfies $\mu(X)_{\pi_\lambda(i)} \geq \mu(X)_{\pi_\lambda(i+1)} + ci^{-\beta}$ for any $X \in \Omega_\lambda$. This implies the probability that $\mu(X)_i \simeq \mu(X)_j$ $(i \neq j)$ is zero.

3. The Definition of State Space Models with Gated Convolution

(Discretized) state space models with input $[u_t]_{t=-L}^{0}$, the latent vectors $[x_t]_{t=-L}^{0}$ and the output $[y_t]_{t=-L}^{0}$ $(u_t \in \mathbb{R}, x_t \in \mathbb{R}, y_t \in \mathbb{R})$ is represented as follows:

$$x_{t+1} = \mathsf{A}x_t + \mathsf{B}u_t, \;\; y_t = \mathsf{C}x_t + \mathsf{D}u_t \;\; (t = -L, \dots, -1),$$

where $\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D} \in \mathbb{R}$ are the (learnable) parameters. Then, the output y_t can be written explicitly as $y_t = \sum_{n=0}^{L+t} \left(\mathsf{C}\mathsf{A}^{t-n}\mathsf{B} + \mathsf{D}\delta_{t-n}\right) u_n$. By setting $h_t := \mathsf{C}\mathsf{A}^t\mathsf{B} + \mathsf{D}\delta_t$ and $h = [h_t]_{t=0}^{L}$, we can rewrite the output as $y_t = (h * u)_t := \sum_{n=0}^{L+t} h_{t-n}u_n$. If the *filter* $[h_t]_{t=0}^{L}$ is precomputed, the output can be computed with $O(L \log L)$ time complexity using FFT algorithm, which is much faster than the computation cost of Transformers, $O(L^2)$.

We consider the state space models with gated convolution like H3 (Fu et al., 2022) and Hyena (Poli et al., 2023). State space model with gated convolution is an architecture that consists of the three components: (i) FNN layers, (ii) gated convolution layers, and (iii) an embedding layer.

(i) FNN layer An FNN with depth L and width W is defined as $f(x) := (A_L\eta(\cdot) + b_L) \circ \cdots \circ (A_1x + b_1)$, where $\eta = \mathrm{ReLU}$, and $A_i \in \mathbb{R}^{d_{i+1}\times d_i}, b_i \in \mathbb{R}^{d_{i+1}}$ with $\max_i d_i \leq W$. Then, we define $\Psi(L, W, S, B)$ as the class of FNNs f with depth L, width W satisfying $\max_i \{\|A_i\|_{\infty}, \|b_i\|_{\infty}\} \leq B$, $\sum_{i=1}^{L} \|A_i\|_0 + \|b_i\|_0 \leq S$.. Note that S and B represent the sparsity and the norm constraint of the parameters, respectively.

(ii) Gated convolution layer Next, we define gated convolution, which was firstly proposed by Dauphin et al. (2017). Let $W_V, W_Q \in \mathbb{R}^{D\times D}$ be learnable weights, and D be the embedding dimension. Then, the gated convolution layer $g\colon \mathbb{R}^{|X|} \to \mathbb{R}^{|X|}$ with window size U is defined as $g(X) := (W^QX) \odot \left(H * (W^VX)\right)$, where $H := \left[c_{1,k} \cos\left(\frac{2\pi j \cdot a_{1,k}}{U}\right) + c_{2,k} \sin\left(\frac{2\pi j \cdot a_{2,k}}{U}\right)\right]_{k,j}$ is a filter controlled by learnable parameters $c_1, c_2, a_1, a_2 \in$

$\mathbb{R}^D$. Note that $\odot\colon \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}^D$ is the elementwise product, and $X * Y := [X_i * Y_i]_{i=1}^d$ for $X = [X_i]_{i=1}^d \in \mathbb{R}^{d\times\infty}, Y = [Y_i]_{i=1}^d \in \mathbb{R}^{d\times\infty}$. Then, we define $\mathcal{C}(U, D, B)$ as the class of gated convolution layer g with window size U, embedding dimension D satisfying $\max\left\{\left\|W^Q\right\|_\infty, \left\|W^V\right\|_\infty, \|c_1\|_\infty, \|c_2\|_\infty\right\} \le B$. This setting is basically inspired Hyena, in which neural networks with $\sin$ activation is used as a convolution filter. However, this can be easily extended to the SSMs with gated convolution with ordenary filter $h_t = \mathsf{C}\mathsf{A}^t\mathsf{B} + \mathsf{D}\delta_t$. See Appendix D for the details. Note that we assume the finite window size U, as in Takakura and Suzuki (2023).

(iii) Embedding layer Finally, we define the embedding layer. For embedding dimension D, an embedding layer is defined as $\mathrm{Emb}(X) = [E_1 X_i + E_2]_{i=-\infty}^{\infty}$, where $E_1 \in \mathbb{R}^{D\times d}$ and $E_2 \in \mathbb{R}^D$ are learnable parameters.

Then, the output of the whole network for input X is computed by $f_M \circ g_M \circ \cdots \circ f_1 \circ g_1 \circ \mathrm{Emb}(X)$. Due to the technical convenience to analyze the estimation error, we consider the setting where the output of the network is bounded. For this purpose, we assume that the output above is fed into the function clip_R defined by $\mathrm{clip}_R(x) := \max\{-R, \min\{x, R\}\}$. Since clip_R can be implemented by the FNN with depth 1 and width 2, such assumption does not far from the practical setting.

Summarizing above, we define $\mathcal{S}(M, U, D, L, W, S, B)$ as the class of data-controlled SSMs F given by

$$F = \mathrm{clip}_R \circ f_M \circ g_M \circ \cdots \circ f_1 \circ g_1 \circ \mathrm{Emb},$$

with $f_i \in \Psi(L, W, S, B)$, $g_i \in \mathcal{C}(U, D, B)$ $(i \in [M])$, and $\|E_1\|_\infty \le B, \|E_2\|_\infty \le B$.

4. Estimation Ability of SSMs with Gated Convolution

First, we show the result for the γ-smooth function class.

Assumption 4.1. *The true function F° is shift-equivariant and satisfies $F_0^\circ \in \mathcal{F}_{p,\theta}^\gamma$, where γ is mixed or anisotropic smoothness. Suppose that it holds $\|F\|_{\mathcal{F}_{p,\theta}^\gamma} \le 1$ and $\|F_0^\circ\|_\infty \le R$, where $R > 0$ is a constant. Additionally, we assume the smoothness parameter a satisfies $\|a\|_{wl^\alpha} \le 1$ for some $0 < \alpha < \infty$ and $a_{ij} = \Omega(\log(|j| + 1))$. Moreover, if γ is mixed smoothness, we assume $\bar{a}_1 < \bar{a}_2$.*

Theorem 4.2. *Suppose that the target function F° satisfies Assumption 4.1. Let $\hat{F}$ be an ERM estimator in $\mathcal{S}(M, U, D, L, W, S, B)$, with M, U, D, L, W, S, B defined as* (F.1) *for $T = \frac{a^\dagger}{2a^\dagger+1}$. Then, for any $l, r \in \mathbb{Z}$, it holds*

$$R_{l,r}(\hat{F}, F^\circ) \lesssim n^{-\frac{2a^\dagger}{2a^\dagger+1}} (\log n)^{6+5/\alpha}.$$

From this, we see that SSMs avoid the curse of dimensionality, and achieve the convergence rate with respect to the sample size that is independent to the dimension of input and output. Additionally, this rate is identical to that of Transformers (Takakura and Suzuki, 2023) up to poly-log factor. Moreover, for the case of anisotropic smoothness with finite dimensional input and output, the convergence rate matches the minimax optimal rate given by Suzuki and Nitanda (2021) up to poly-log factor.

Next, we evaluate the estimation ability of SSMs for piecewise γ-smooth functions.

Assumption 4.3. *The true function F° is shift-equivariant and satisfies $F_0^\circ \in \mathcal{P}_{p,\theta}^\gamma$, where γ is mixed or anisotropic smoothness. Additionally, we suppose that $\|F_0^\circ\|_{\mathcal{P}_{p,\theta}^\gamma} \le 1$ and $\|F_0^\circ\|_\infty \le R$, where $R > 0$ is a constant. We also assume $a_{ij} = \Omega(j^\alpha)$ and $\|a\|_{wl^\alpha} \le 1$ for some $0 < \alpha < \infty$. Moreover, there exists a importance function μ of $\mathcal{P}_{p,\theta}^\gamma$ that belongs to γ-smooth function class. For case (i), we further assume that μ satisfies Assumption 4.1.*

Theorem 4.4. *Suppose that the target function F° satisfies Assumption 4.3. Let $\hat{F}$ be an ERM estimator in $\mathcal{S}(M, U, D, L, W, S, B)$, with M, U, D, L, W, S, B defined as* (F.2) *for $T = \frac{a^\dagger}{2a^\dagger+1}$. Then, for any $l, r \in \mathbb{Z}$, it holds*

$$R_{l,r}(\hat{F}, F^\circ) \lesssim n^{-\frac{2a^\dagger}{2a^\dagger+1}} \cdot \mathrm{polylog}(n, V).$$

As well as the case of γ-smooth functions, this convergence rate with respect to n matches that of Transformers. This indicates that SSMs have ability to select important tokens depending on the inputs, similary to Transformers. Moreover, since the estimation error bound depends on V with only poly-log factor, if $V = \mathrm{poly}(n)$, the estimation error rate does not change up to poly-log factor. This aspect also matches the result of Transformers.

In addition to the setting above, we consider the functions whose smoothness structure also changes depending on the output token. This setting is inspired by the ability of Transformers to solve the task called associative recall (Ba et al., 2016). We show that SSMs can also replace Transformers in this setting. See Appendix B for the detailed results.

We conclude that SSMs are comparable to Transformers in terms of estimation ability, when (i) the target function has a smoothness structure and (ii) the positions of important tokens depend on the input.

5. Conclusion

We theoretically explored the possibility of SSMs as an alternative to Transformers, and showed the approximation and estimation ability of SSMs for certain classes. Consequently, we proved that SSMs have the same estimation ability as Transformers for those function classes.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

C. A. Alonso, J. Sieber, and M. N. Zeilinger. State space models as foundation models: A control theoretic overview. *arXiv preprint arXiv:2403.16899*, 2024.

J. Ba, G. E. Hinton, V. Mnih, J. Z. Leibo, and C. Ionescu. Using fast weights to attend to the recent past. *Advances in neural information processing systems*, 29, 2016.

M. Chen, H. Jiang, W. Liao, and T. Zhao. Nonparametric regression on low-dimensional manifolds using deep relu networks: Function approximation and statistical recovery. *Information and Inference: A Journal of the IMA*, 11 (4):1203–1253, 2022.

N. M. Cirone, A. Orvieto, B. Walker, C. Salvi, and T. Lyons. Theoretical foundations of deep selective state-space models. *arXiv preprint arXiv:2402.19047*, 2024.

Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. Language modeling with gated convolutional networks. In *International conference on machine learning*, pages 933–941. PMLR, 2017.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

D. Y. Fu, T. Dao, K. K. Saab, A. W. Thomas, A. Rudra, and C. Re. Hungry hungry hippos: Towards language modeling with state space models. In *The Eleventh International Conference on Learning Representations*, 2022.

K. Goel, A. Gu, C. Donahue, and C. Ré. It's raw! audio generation with state-space models. In *International Conference on Machine Learning*, pages 7616–7633. PMLR, 2022.

K. Grešová, V. Martinek, D. Čechák, P. Šimeček, and P. Alexiou. Genomic benchmarks: a collection of datasets for genomic sequence classification. *BMC Genomic Data*, 24(1):25, 2023.

A. Gu and T. Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

A. Gu, K. Goel, and C. Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2021.

M. Imaizumi and K. Fukumizu. Deep neural networks learn non-smooth functions effectively. In *The 22nd international conference on artificial intelligence and statistics*, pages 869–878. PMLR, 2019.

S. Massaroli, M. Poli, D. Fu, H. Kumbong, R. Parnichkun, D. Romero, A. Timalsina, Q. McIntyre, B. Chen, A. Rudra, et al. Laughing hyena distillery: Extracting compact recurrences from convolutions. *Advances in Neural Information Processing Systems*, 36, 2024.

W. Merrill, J. Petty, and A. Sabharwal. The illusion of state in state-space models. *arXiv preprint arXiv:2404.08819*, 2024.

R. Nakada and M. Imaizumi. Adaptive approximation and generalization of deep neural network with intrinsic dimensionality. *Journal of Machine Learning Research*, 21 (174):1–38, 2020.

E. Nguyen, M. Poli, M. Faizi, A. Thomas, M. Wornow, C. Birch-Sykes, S. Massaroli, A. Patel, C. Rabideau, Y. Bengio, et al. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *Advances in neural information processing systems*, 36, 2024.

K. Oko, S. Akiyama, and T. Suzuki. Diffusion models are minimax optimal distribution estimators. In *International Conference on Machine Learning*, pages 26517–26582. PMLR, 2023.

S. Okumoto and T. Suzuki. Learnability of convolutional neural networks for infinite dimensional input via mixed and anisotropic smoothness. In *International Conference on Learning Representations*, 2021.

D. Perekrestenko, P. Grohs, D. Elbrächter, and H. Bölcskei. The universal approximation power of finite-width deep relu networks. *arXiv preprint arXiv:1806.01528*, 2018.

P. Petersen and F. Voigtlaender. Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, 108:296–330, 2018.

M. Poli, S. Massaroli, E. Nguyen, D. Y. Fu, T. Dao, S. Baccus, Y. Bengio, S. Ermon, and C. Ré. Hyena hierarchy: Towards larger convolutional language models. *arXiv preprint arXiv:2302.10866*, 2023.

A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR, 2023.

G. Saon, A. Gupta, and X. Cui. Diagonal state space augmented transformers for speech recognition. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

J. Schmidt-Hieber. Nonparametric regression using deep neural networks with relu activation function. 2020.

T. Suzuki. Adaptivity of deep relu network for learning in besov and mixed smooth besov spaces: optimal rate and curse of dimensionality. In *International Conference on Learning Representations*, 2018.

T. Suzuki and A. Nitanda. Deep learning is adaptive to intrinsic dimensionality of model smoothness in anisotropic besov space. *Advances in Neural Information Processing Systems*, 34:3609–3621, 2021.

S. Takakura and T. Suzuki. Approximation and estimation ability of transformers for sequence-to-sequence functions with infinite dimensional input. In *Proceedings of the 40th International Conference on Machine Learning*, pages 33416–33447. PMLR, 2023.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

S. Wang and B. Xue. State-space models with layer-wise nonlinearity are universal approximators with exponential decaying memory. *Advances in Neural Information Processing Systems*, 36, 2024.

A. Other Related Works

Estimation abilities of deep neural networks. Leaving aside SSMs, many studies have investigated the abilities of deep neural networks to estimate functions. Some of them analyze the estimation abilities of fully connected neural networks (FNNs) with the assumption that the target function is in certain function classes (Schmidt-Hieber, 2020; Suzuki, 2018) or have a specific smoothness structure (Suzuki and Nitanda, 2021). Moreover, Nakada and Imaizumi (2020) and Chen et al. (2022) consider the setting that the data distribution has a low-dimensional structure. Additionally, Okumoto and Suzuki (2021) studied convolutional neural networks (CNNs) and showed that CNNs can estimate the functions that have smoothness structures with the minimax optimal rate even if the input is infinite-dimensional. As for the Transformers, Takakura and Suzuki (2023) showed that Transformers can estimate the functions with infinite-dimensional input as well as CNNs. Additionally, they showed that Transformers can estimate the functions whose smoothness structure changes depending on the input.

The function classes with piecewise smoothness are also considered in Petersen and Voigtlaender (2018) and (Imaizumi and Fukumizu, 2019). They do not consider anisotropic smoothness or the sequence-to-sequence functions, while we consider such situations.

Abilities of state space models. Some studies have showed the limitations of state space models. For example, Merrill et al. (2024) show that SSMs cannot solve sequential problems from the view of computational complexity theory. Additionally, Gu and Dao (2023) pointed out that SSMs are less effective for the tasks to handle discrete and information-dense data such as language processing.

There are some other directions to investigate the abilities of SSMs. For example, Alonso et al. (2024) summaried the features of the existing SSM-based architectures. Instead of focusing on the statistical aspect, they mainly provide a comprehensive understanding of existing SSMs. Moreover, Cirone et al. (2024) studied the abilities of SSMs using rough path theory. This study discuss the limtations of SSMs, but they did not give comparison between Transformers and SSMs.

Other architectures with SSMs. Gu and Dao (2023) proposed an SSM-based architecture called Mamba, whose filter is controlled by the input. While convolution with FFT algorithms cannot be used for Mamba, they proposed a hardware-aware efficient implementation. In this paper, we do not focus on the setting that filters are controlled by the input, and we consider SSMs with gated convolution with data-independent filters.

B. Additional Results

As for the importance function μ, Takakura and Suzuki (2023) assume that it belongs to γ-smooth function class. Consequently, the value of importance of a token is unique in the input sequence, and does not change depending on the position of the output token. In order to consider the setting *where the importance of tokens can change depending on the output token*, in addition to the case where μ is **(i) γ-smooth**, we consider the following case:

(ii) similarity of features: Let $\zeta_1, \dots, \zeta_{d'} \in \mathcal{F}^{\gamma}_{p,\theta}$, and $v_j \coloneqq [\zeta_i(\Sigma_j X)]_{i=1}^{d'} \in \mathbb{R}^{d'}$ $(j \in \mathbb{N})$. Suppose that $\|v_j\|_2 \leq 1$ for any $j \in \mathbb{N}$. Then, the importance function μ is can be represented by $\mu(X) = \left[-\|v_0 - v_j\|_2\right]_{j=-\infty}^{0}$ or $\mu(X) = \left[v_0^\top v_j\right]_{j=-\infty}^{0}$.

In this setting, v_j intuitively represents the features of the token at the position j, and we assume that the importance of a token is controlled by the similarity between the target token and the current token.

The setting (ii) makes it possible for us to consider additional synthetic task. Specifically, the functions in setting (ii) includes the task called associative recall Ba et al. (2016). In this task, the query token has appeared in the past, and the model is required to output the same token that followed it in the previous occurrence, e.g., if the input is "`a 2 c 4 b 3 d 1 e 7 c`", the model should output "`4`". Additionally, (ii) includes in-context learning with k-nearest neighbors algorithm.

Note that the setting (i) includes some important settings such as induction head and selective copying Gu and Dao (2023). Indeed, in those tasks, the absolute position of the important token is fixed.

For the setting (ii), we have the following theorem, whose convergence rate is the same as Theorem 4.4.

Theorem B.1. *Suppose that the target function F° satisfies Assumption 4.3. Additionally, the importance function μ given by (ii) similarity of features. Let $\hat{F}$ be an ERM estimator in $\mathcal{S}(M, U, D, L, W, S, B)$, with M, U, D, L, W, S, B defined as*

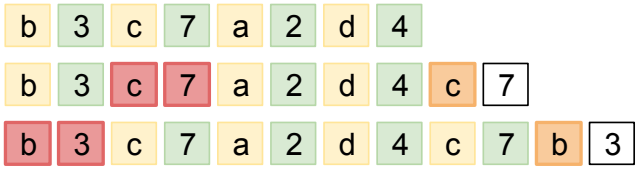

Figure B.1. Conceptual illustrations for piecewise γ-smooth function class with similarity-based importance. This illustrate the task to predict the next token in the sequence. In this task, models have to output the token associated with the last token. We also prove that SSMs can solve such tasks, i.e., SSMs can extract appropriate tokens even if the position of essential tokens are *different for each token in the same sequence*.

(F.2) *for* $T = \frac{a^\dagger}{2a^\dagger+1}$. *Then, for any* $l, r \in \mathbb{Z}$, *it holds*

$$R_{l,r}(\hat{F}, F^\circ) \lesssim n^{-\frac{2a^\dagger}{2a^\dagger+1}} \cdot \mathrm{polylog}(n, V).$$

This fact reveals that, even in situations where the positions of important tokens differ for each output token, it is possible for SSMs to identify them.

C. Numerical Experiments

To demonstrate that our theory is compatible with the real-world tasks, we conducted two numerical experiments. We utilize the pretrained model of Hyena provided by Nguyen et al. (2024), which is trained via the next token prediction task with the nucleic acid base sequences. We fine-tune the model using Genomic Benchmark dataset (Grešová et al., 2023). Additional description on the experimental setting can be found in Appendix K.

First, we verify that SSMs can select important tokens depending on the input. We fine-tune the model with the binary classification task. Then, we choose one sequence in the test data and plot the transition of the probability of correct classification when we repeatedly mask the input tokens that do not affect the classification result. Additionally, we observe how the correctness for other sequences changes when we mask the same tokens as the chosen sequence.

The result is shown on the left of Figure C.1. We can see that SSMs can classify the sequences even if we mask most of the tokens. This indicates that the number of important tokens to classify the sequence is small. Moreover, the accuracies for other sequences decrease when there are a few tokens left, which means that the important tokens of other sequences differ from the chosen sequence.

Second, we demonstrate the ability of SSMs to extract important tokens depending on the output token. We consider the task to predict the masked token at the last of sequence. Then, we choose one token for each sequence and plot the transition of the probability of correct prediction when we repeatedly mask the tokens whose impact on the prediction is small. Additionally, we plot the transition of the correctness for other tokens when we mask the the tokens at the same positions.

We show the result in the right part of Figure C.1. We can notice that the accuracy for the chosen token decreases slower than the other tokens. This reveals that the important tokens to pay attention to are different depending on the position of the output token, and SSMs can adaptively extract the important tokens depending on the output token.

D. Extension to ordinary SSM filter

In this section, we describe how to extend our setting to the ordinary SSM filter. More specifically, our setting with embedding dimension D can be extended to the ordinary SSM filter with embedding dimension $4D$.

For simplicity, we consider the case $D = 1$. We constuct the parameters $\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D} \in \mathbb{R}^{2\times 2}$ to make the filter $h_t := \mathsf{C}\mathsf{A}^t\mathsf{B} + \mathsf{D}\delta_{t-n}$ same as the filter defined in Section 3. Let us set $\mathsf{D} = 0$, and

$$\mathsf{A} = \begin{bmatrix} \cos\left(\frac{2\pi a_{1,1}}{U}\right) & -\sin\left(\frac{2\pi a_{1,1}}{U}\right) & 0 & 0 \\ \sin\left(\frac{2\pi a_{1,1}}{U}\right) & \cos\left(\frac{2\pi a_{1,1}}{U}\right) & 0 & 0 \\ 0 & 0 & \cos\left(\frac{2\pi a_{1,2}}{U}\right) & -\sin\left(\frac{2\pi a_{1,2}}{U}\right) \\ 0 & 0 & \sin\left(\frac{2\pi a_{1,2}}{U}\right) & \cos\left(\frac{2\pi a_{1,2}}{U}\right) \end{bmatrix}.$$

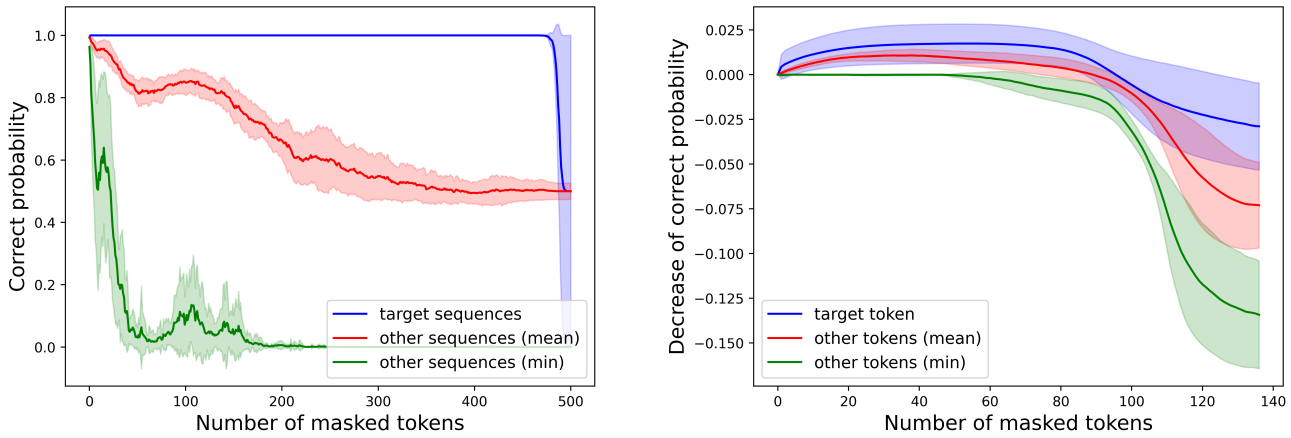

Figure C.1. The experimental results. Left: We choose one target sequence from the test data, and repeatedly mask the tokens with less importance. Then, the probability of correct classification decreases slowly, which means the essential tokens for the classification are limited. On the other hand, if we mask tokens at the same position for other sequences, the accuracy decreases faster. This indicates that the model can select the important tokens depending on the input. Right: We fix one target token for each sequence, and repeatedly mask the input tokens with less importance. Then, the probability of correct prediction decreases slower for the chosen token than the other tokens. This reveals that the important tokens to pay attention is different depending on the output token.

Then, we have

$$\mathsf{A}^t = \begin{bmatrix} \cos\left(\frac{2\pi a_{1,1} t}{U}\right) & -\sin\left(\frac{2\pi a_{1,1} t}{U}\right) & 0 & 0 \\ \sin\left(\frac{2\pi a_{1,1} t}{U}\right) & \cos\left(\frac{2\pi a_{1,1} t}{U}\right) & 0 & 0 \\ 0 & 0 & \cos\left(\frac{2\pi a_{1,2} t}{U}\right) & -\sin\left(\frac{2\pi a_{1,2} t}{U}\right) \\ 0 & 0 & \sin\left(\frac{2\pi a_{1,2} t}{U}\right) & \cos\left(\frac{2\pi a_{1,2} t}{U}\right) \end{bmatrix}.$$

Therefore, if we set

$$\mathsf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathsf{C} = \begin{bmatrix} c_{1,1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ c_{1,2} & 0 & 0 & 0 \end{bmatrix},$$

then we have

$$h_t = \begin{bmatrix} c_{1,1} \cos\left(\frac{2\pi a_{1,1} t}{U}\right) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ c_{1,2} \sin\left(\frac{2\pi a_{1,2} t}{U}\right) & 0 & 0 & 0 \end{bmatrix}.$$

Then, if we appropriately set W^V and W^Q, this filter can realize the same output with our setting.

While we do not show the estimation ability for the filter above, we can easily extend our proof to derive the almost same estimation error bound for it.

E. Key Insight on SSMs with Gated Convolution

Before starting the proof, we show the key insight on SSMs with gated convolution.

Suppose $f\colon \mathbb{R}^d \to \mathbb{R}^{2n}, x \mapsto \left[\phi_A^{(1)}(x), \ldots, \phi_A^{(n)}(x), \phi_B^{(1)}(x), \ldots, \phi_B^{(n)}(x)\right]$ is a fully connected neural network. Additionally, let $g \in \mathcal{C}(U, D, B)$ with $D = 2n$. Recall that g is defined as

$$g(X) = \left(W^Q X\right) \odot \left(H * \left(W^V X\right)\right) \quad (X \in \mathbb{R}^{2n \times \infty}),$$

where $H \in \mathbb{R}^{2n \times 2n}$ is a filter defined by

$$H_{k,j} \coloneqq c_{1,k} \cos\left(\frac{2\pi j \cdot a_{1,k}}{U}\right) + c_{2,k} \sin\left(\frac{2\pi j \cdot a_{2,k}}{U}\right).$$

Let us set the parameters $W^Q, W^V \in \mathbb{R}^{2n \times 2n}, c_1, c_2 \in \mathbb{R}^{2n}$ in g as

$$\begin{aligned} W^Q &= \begin{bmatrix} O & I \\ O & O \end{bmatrix}, \quad W^V = \begin{bmatrix} I & O \\ O & O \end{bmatrix}, \\ c_1 &= \begin{bmatrix} \alpha_1 \cdot \mathbb{I}_S(1) & \cdots & \alpha_n \cdot \mathbb{I}_S(n) \end{bmatrix}^\top, \\ c_2 &= \begin{bmatrix} \alpha_1 \cdot \mathbb{I}_{\bar{S}}(1) & \cdots & \alpha_n \cdot \mathbb{I}_{\bar{S}}(n) \end{bmatrix}^\top, \end{aligned}$$

where $S \subseteq [n]$ and $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$. Then, for any input $X \coloneqq [x_t]_{t=-\infty}^0 \in \mathbb{R}^{2n \times \infty}$, we have

$$(g \circ f(X))_0 = \begin{bmatrix} \sum_{t=-U}^0 \alpha_1 \cdot \psi^{(1)}(t/U) \phi_A^{(1)}(x_t) \phi_B^{(1)}(x_0) \\ \vdots \\ \sum_{t=-U}^0 \alpha_n \cdot \psi^{(n)}(t/U) \phi_A^{(n)}(x_t) \phi_B^{(n)}(x_0) \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

where

$$\psi^{(k)}(s) \coloneqq \begin{cases} \cos(2\pi a_{1,k} \cdot s) & (k \in S), \\ \sin(2\pi a_{2,k} \cdot s) & (k \notin S). \end{cases}$$

Therefore, we can see that, for any function $K \colon \mathbb{R}^{2n+1} \to \mathbb{R}$ that can be approximated by the form of

$$K(t, x, y) = \sum_{k=1}^n \alpha_k \cdot \psi^{(k)}(t/U) \phi_A^{(k)}(x) \phi_B^{(k)}(y),$$

one FNN and one SSM with gated convolution can approximate its summation over

$$(t, x, y) = (0, x_0, x_0), (-1, x_{-1}, x_0), \ldots, (-U, x_{-U}, x_0),$$

i.e.,

$$\sum_{t=-U}^0 K(t, x_t, x_0).$$

In the following proof, we use this fact for sevaral times as described below.

Extracting a token at a specific position We can extract one token from $x_0, x_{-1}, \ldots, x_{-U}$ by setting K to satisfy

$$K(t, x, y) \approx \begin{cases} x & (t = t_0), \\ 0 & (t \neq t_0), \end{cases}$$

for some $t_0 \in [-U : 0]$.

Extracting a feature of the token with high similarity Let $\rho \colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be a kernel function that measures the similarity between two vectors. Suppose that ρ can be approximated by a finite sum as follows:

$$\rho(x, y) \approx \sum_{k=1}^n \alpha_i \cdot \eta_k(x) \eta_k(y).$$

Let us set K to satisfy

$$K(t,x,y) \approx \sum_{k=1}^{n} x_m \cdot \eta_k(x) \cdot \eta_k(y),$$

for some $m \in [n]$. Then we can extract the m-th coordinate of the token that has the highest similarity with x_0 among $x_{-U}, \ldots, x_0$, if the similarities between other tokens and x are significantly lower. Note that $x \mapsto x_m \cdot \eta_k(x)$ can be approximated by a FNN.

F. Approximation Ability of SSMs with Gated Convolution

To establish the estimation abilities of SSMs, we first show the results on the approximation abilities of SSMs for γ-smooth functions and piecewise γ-smooth functions.

F.1. Approximation of γ-smooth functions

Now, we establish the approximation ability of γ-smooth functions via SSMs. First, we state the detailed assumptions on the target function.

Remark. The assumption $\|a\|_{wl^\alpha} \leq 1$ implies that the j-th smallest element of smoothness parameter a increases polynomially with respect to j, which indicates the sparsity of the important features. Moreover, the assuption $a_{ij} = \Omega(\log(|j|+1))$ means that the token distant from the current token is less important. Thus, those two assumptions are natural in various applications.

Then, we have the following theorem which shows that the SSM can approximate γ-smooth functions.

Theorem F.1. *Suppose that target function F° satisfies Assumption 4.1. Then, for any $T > 0$, there exists an SSM $F \in \mathcal{S}(M,U,D,L,W,S,B)$ with*

$$\begin{aligned} &M = 1, \quad \log U \sim T, \quad D \sim T^{1/\alpha}, \quad L \sim T, \quad W \sim T^{1/\alpha}, \\ &W' \sim T^{1/\alpha} 2^{T/a^\dagger}, \quad S \sim T^{2/\alpha} \max\left\{T^{2/\alpha}, T^2\right\} 2^{T/a^\dagger}, \quad \log B \sim T^{1/\alpha}, \end{aligned} \tag{F.1}$$

such that $\|F - F^\circ\|_{2,P_X} \lesssim 2^{-T}$.

The proof can be found in Appendix H. We can see that, the number of parameters to achieve certain error does not suffer from the infinite dimensionality. This is due to the fact the SSMs use same convolution filter for each tokens, as well as self-attention of Transformers.

F.2. Approximation of piecewise γ-smooth functions

Next, we show that the SSMs can approximate the piecewise γ-smooth functions. To this end, we first state the detailed assumptions on the target function.

Remark. The assumptions on the norm of the smoothness parameter a and the function F_0° are same as Assumption 4.1. The interpretion for the conditions on the importance function is described in Appendix B. The condition $a_{ij} = \Omega(j^\alpha)$ indicates that the function is smoother with respect to the token with small importance the tokens are sorted in order of decreasing importance by the map Π. Note that, unlike the case of Assumption 4.1, the condition $a_{ij} = \Omega(j^\alpha)$ is imposed on the smoothness of permuted tokens, and j can be different from the original position.

Then, we have the following theorem on the approximation ability of SSMs for piecewise γ-smooth functions.

Theorem F.2. *Let F° be a function satisfying Assumption 4.3. Then, for any $T > 0$, there exists a SSM $\hat{F} \in \mathcal{S}(M,U,D,L,W,S,B)$ with*

$$\begin{aligned} &M \lesssim T^{1/\alpha}, \quad U = V, \quad D \lesssim T^{c_{\alpha,\beta}} \log^2 V, \quad L \lesssim T^{c_{\alpha,\beta}} \log^3 V, \\ &W \lesssim 2^{T/a^\dagger} T^{c_{\alpha,\beta}} \log^2 V, \quad S \lesssim 2^{T/a^\dagger} T^{c_{\alpha,\beta}} \log^3 V, \quad \log B \lesssim T^{c_{\alpha,\beta}} \log^3 V, \end{aligned} \tag{F.2}$$

such that $\left\|F^\circ - \hat{F}\right\|_2 \lesssim 2^{-T}$. Here, $c_{\alpha,\beta}$ is a constant depending on α and β such that $c_{\alpha,\beta} \leq 4 + 2/\alpha + 3\beta/\alpha$.

The proof can be found in Appendix I. This result reveals that the number of parameters to attain the error 2^{-T} is same as the case of γ-smoothness, which implies SSMs have the ability to extract tokens depending on the input sequence and the token at the output position.

G. Auxiliary Lemmas

In the following discusstion, to simplify the notation, we define the function class $\Psi'(D, B)$ by

$$\Psi'(D, B) \coloneqq \Big\{ t \mapsto [c_{1,k} \cos(2\pi a_{1,k} t) + c_{2,k} \sin(2\pi a_{2,k} t)]_{k=1}^{D} \;\Big|\; \|c\|_\infty \le B, \|a\|_\infty \le B. \Big\}.$$

First, we prove the following lemma, which states the properties of the $\mathrm{Softmax}$ and multi-variate Swish function.

Lemma G.1 (Properties of $\mathrm{Softmax}$ and Multi-variate Swish function)**.** *Fix $\theta \in \mathbb{R}^d$. Assume that there exists an index $i^* \in [d]$ and $\delta > 0$ such that $\theta_{i^*} > \theta_i + \delta$ for all $i \neq i^*$. Then, the following two statements hold:*

1. *(Lemma C.1 of Takakura and Suzuki (2023)) It holds*

$$\sum_{i=1}^{d} |\mathrm{Softmax}(\theta)_i - \delta_{i,i^*}| \le 2d \exp(-\delta).$$

2. *For any $x \in [0,1]^d$, it holds*

$$\left| \sum_{i=1}^{d} \mathrm{Softmax}(\theta)_i \cdot x_i - \max\{x_1, \dots, x_d\} \right| \le 2d^2 \exp(-\delta).$$

Proof. We prove the second one. Using the first argument, we have

$$\begin{aligned}
&\left| \sum_{i=1}^{d} \mathrm{Softmax}(\theta)_i \cdot x_i - \max\{x_1, \dots, x_d\} \right| \\
&\le \left| \sum_{i \neq i^*} \mathrm{Softmax}(\theta)_i \cdot x_i + (\mathrm{Softmax}(\theta)_{i^*} \cdot x_{i^*} - \max\{x_1, \dots, x_d\}) \right| \\
&= \left| \sum_{i \neq i^*} \mathrm{Softmax}(\theta)_i \cdot x_i + (\mathrm{Softmax}(\theta)_{i^*} \cdot x_{i^*} - \delta_{i,i^*} x_{i^*}) \right| \\
&\le \sum_{i \neq i^*} \mathrm{Softmax}(\theta)_i \cdot x_i + |\mathrm{Softmax}(\theta)_{i^*} - \delta_{i,i^*}| \cdot x_{i^*} \\
&\le \sum_{i=1}^{d} |\mathrm{Softmax}(\theta)_i - \delta_{i,i^*}| \cdot x_i \\
&\le 2d^2 \exp(-\delta),
\end{aligned}$$

which completes the proof. □

The following lemma shows the approximation ability of FNN for some elementary functions.

Lemma G.2 (Lemma F.6, Lemma F.7, Lemma F.12 of Oko et al. (2023), Corollary 4.2 of Perekrestenko et al. (2018))**.** *The following statements hold:*

1. *Let $d \ge 2, C \ge 1, \epsilon_{\mathrm{error}} \in (0, 1]$. For any $\epsilon > 0$, there exists a neural network $f_{\mathrm{mult}} \in \Psi(L, W, S, B)$ with*

$$L \lesssim (\log \epsilon^{-1} + d \log C) \cdot \log d, \quad W \lesssim d, \quad S \lesssim d \log \epsilon^{-1} + d \log C, \quad \log B \lesssim d \log C,$$

such that, for any $x \in [0, C]^d$ and $x \in \mathbb{R}^d$ with $\|x - x'\|_\infty \leq \epsilon_{\mathrm{error}}$, it holds

$$\left| f_{\mathrm{mult}}(x') - \prod_{i=1}^{d} x_i \right| \leq \epsilon + d \cdot C^d \cdot \epsilon_{\mathrm{error}}.$$

2. *For any $\epsilon \in (0, 1)$, there exists $f_{\mathrm{rec}} \in \Psi(L, W, S, B)$ with*

$$L \lesssim \log^2 \epsilon^{-1}, \quad W \lesssim \log^3 \epsilon^{-1}, \quad S \lesssim \log^4 \epsilon^{-1}, \quad \log B \lesssim \log \epsilon^{-1},$$

such that, for any $x \in [\epsilon, \epsilon^{-1}]$ and $x' \in \mathbb{R}$, it holds

$$\left| f_{\mathrm{rec}}(x') - \frac{1}{x} \right| \leq \epsilon + \frac{|x' - x|}{\epsilon^2}.$$

3. *For any $\epsilon > 0$, there exists $f_{\exp} \in \Psi(L, W, S, B)$ with*

$$L \lesssim \log^2 \epsilon^{-1}, \quad W \lesssim \log \epsilon^{-1}, \quad S \lesssim \log^2 \epsilon^{-1}, \quad \log B \lesssim \log^2 \epsilon^{-1},$$

such that, for any $x, x' \geq 0$, it holds

$$|f_{\exp}(x') - \exp(x)| \leq \epsilon + |x' - x|.$$

4. *For any $\epsilon > 0, a > 0, b \in \mathbb{R}, C \geq 1$, there exists $f_{\cos} \in \Psi(L, W, S, B)$ with*

$$\begin{aligned} &L \lesssim \log^2 \epsilon^{-1} + \log(aD + b), \quad W \lesssim 1, \\ &S \lesssim \log^2 \epsilon^{-1} + \log(aD + b), \quad \log B \lesssim \max\{1, \log |b/a|\}, \end{aligned}$$

such that, for any $x \in [-D, D]$, it holds

$$|f_{\cos}(x) - \cos(ax + b)| \leq \epsilon.$$

The following is a famous fact that there exists a neural network that realize the clipping function.

Lemma G.3. *Let $a, b \in \mathbb{R}$. There exists a neural neural network $f_{\mathrm{clip}} \in \Psi(L, W, S, B)$ with*

$$L \lesssim 1, \quad W \lesssim 1, \quad S \lesssim 1, \quad B \lesssim |a| + |b|,$$

such that, for any $x \in \mathbb{R}$, it holds

$$f_{\mathrm{clip}}(x) = \begin{cases} a & \text{if } x \leq a, \\ x & \text{if } a \leq x \leq b, \\ b & \text{if } b \leq x. \end{cases}$$

Lastly, we state the following lemma, which shows that the dirac delta function can be approximated by a neural network.

Lemma G.4. *There exists $N \in \mathbb{N}$ and FNNs $\phi_n, \phi_n', \phi_n'' \in \Psi_{1,1}(L, W, S, B)$, $\psi_n, \psi_n' \in \Psi'(1, B)$ $(n = 1, \ldots, N)$ with*

$$\begin{aligned} &N \lesssim \log \epsilon^{-1}, \\ &L \lesssim \log^2 \epsilon^{-1} \log^2 \kappa, \quad W \lesssim \log^2 \epsilon^{-1}, \quad S \lesssim \log^4 \epsilon^{-1} \log^2 \kappa, \quad \log B \lesssim \log \epsilon^{-1} \log \kappa, \\ &L' = 1, \quad W' \lesssim \log^2 \epsilon^{-1}, \quad S' \lesssim \log^2 \epsilon^{-1}, \end{aligned}$$

such that,

- *for any $t, x \in \mathbb{R}$, it holds*

$$\left| \exp\Big(-\kappa \cdot \sin^2\Big(\frac{\pi}{2}(t - x)\Big)\Big) - \sum_{n=1}^{N} \psi_n(t) \phi_n(x) \right| \lesssim \epsilon,$$

- *for any $x, y \in \mathbb{R}$, it holds*

$$\left|\exp\Big(-\kappa \cdot \sin^2\left(\frac{\pi}{2}(x-y)\right)\Big) - \sum_{n=1}^{N} \phi_n'(x)\phi_n''(y)\right| \lesssim \epsilon,$$

- *for any $t \in [-1, 1]$, it holds*

$$\left|\exp\left(-\kappa \cdot \sin^2\left(\frac{\pi t}{2}\right)\right) - \sum_{n=1}^{N} \psi_n'(t)\right| \lesssim \epsilon.$$

Proof. The first part of the proof is inspired by Lemma F.12 of Oko et al. (2023). Let us set $A = \log 3\epsilon^{-1}$. The Taylor expansion of $\exp$ shows that, for any $x \in [0, A]$, it holds

$$\left|\exp(-x) - \sum_{n=0}^{N-1} \frac{(-1)^n}{n!} x^n\right| \le \frac{A^N}{N!}.$$

Additionally, we can evaluate the right-hand side as $A^k/k! \le (\mathrm{e}A/k)^k$. Therefore, if we set $N = \max\left\{2\mathrm{e}A, \lceil \log_2 3\epsilon^{-1} \rceil\right\}$, the error can be bounded by $\epsilon/3$. Moreover, for $x > A$, we have

$$\begin{aligned}\left|\exp(-x) - \sum_{n=0}^{N-1} \frac{(-1)^n}{n!} x^n\right| &\le |\exp(-x) - \exp(-A)| + \left|\exp(-A) - \sum_{n=0}^{N-1} \frac{(-1)^n}{n!} x^n\right| \\ &\le \frac{\epsilon}{3} + \frac{2\epsilon}{3} = \epsilon.\end{aligned}$$

Next, let us approximate $\displaystyle\sum_{n=0}^{N-1} \frac{(-\kappa)^n}{n!} \sin^{2n}\left(\frac{\pi}{2}(t-x)\right)$. We use the fact that

$$\begin{aligned}\sin^{2n}(x) = \left(\frac{\mathrm{e}^{ix} - \mathrm{e}^{-ix}}{2}\right)^{2n} &= \frac{1}{2^{2n}} \sum_{k=0}^{2n} \binom{2n}{k} (-1)^k \mathrm{e}^{i(2k-2n)x} \\ &= \frac{(-1)^n}{2^{2n}}\binom{2n}{n} + \sum_{k \ge n+1} \frac{(-1)^k}{2^{2n-1}} \binom{2n}{k} \cos\left((2k-2n)x\right),\end{aligned}$$

where $c_n = 1$ if n is even and $c_n = 0$ if n is odd. Thus, we have

$$\begin{aligned}&\sum_{n=0}^{N-1} \frac{(-\kappa)^n}{n!} \sin^{2n}\left(\frac{\pi}{2}(t-x)\right) \\ &\quad= \sum_{n=0}^{N-1} \frac{\kappa^n}{n! 2^{2n}} \binom{2n}{n} + \sum_{n=0}^{N-1} \sum_{k \ge n+1} \frac{(-\kappa)^n}{n!} \frac{1}{2^{2n-1}} \binom{2n}{k} \cos\left(\pi(k-n)(t-x)\right) \\ &\quad= \sum_{n=0}^{N-1} \frac{\kappa^n}{n! 2^{2n}} \binom{2n}{n} + \sum_{n=0}^{N-1} \sum_{k \ge n+1} \frac{(-\kappa)^n}{n!} \frac{1}{2^{2n-1}} \binom{2n}{k} \Big(\cos\left(\pi(k-n)t\right)\cos\left(\pi(k-n)x\right) \\ &\qquad\qquad + \sin\left(\pi(k-n)t\right)\sin\left(\pi(k-n)x\right)\Big),\end{aligned}$$

which is decomposed into the sum of products of functions of t and x. Since

$$\left|\frac{(-\kappa)^n}{n!} \frac{1}{2^{2n-1}} \binom{2n}{k}\right| \le \frac{\kappa^n}{n! 2^n} \frac{(2n)!}{k!(2n-k)!} \le \frac{\kappa^n}{n! 2^n} \frac{2^n (n!)^2}{(\max(k, 2n-k))!} = \frac{\kappa^n}{n! 2^n} \frac{2^n (n!)^2}{n!} \le \kappa^N,$$

we can see that, there exists $C_0, C_{n,k}$ ($n \in [N], k \in [0 : n/2]$) with

$$C_0 \le \kappa^N, \quad C_{n,k} \le \kappa^N,$$

such that

$$\sum_{n=0}^{N-1} \frac{(-\kappa)^n}{n!} \sin^{2n}\left(\frac{\pi}{2}(t-x)\right) = C_0 + \sum_{n=0}^{N-1} \sum_{k\geq (n+1)/2} C_{n,k}\big(\cos\left(\pi(k-n)t\right)\cos\left(\pi(k-n)x\right) + \sin\left(\pi(k-n)t\right)\sin\left(\pi(k-n)x\right)\big).$$

The second equation to be proved is already obtained setting $x = 0$.

Finally, we approximate each term using neural networks. Lemma G.2 implies that, for any n, k and $\epsilon > 0$, there exists a neural network $\phi_{1,n,k}, \phi_{2,n,k} \in \Psi_{1,1}(L, W, S, B)$ with

$$L \lesssim N^2 \log^2 \kappa + \log^2 \epsilon^{-1}, \quad W \lesssim 1, \quad S \lesssim N^2 \log^2 \kappa + \log^2 \epsilon^{-1}, \quad \log B \lesssim 1,$$

such that

$$|\cos(\pi(k-n)x) - \phi_{1,n,k}(x)| \leq \epsilon/(N^2\kappa^N), \quad |\sin(\pi(k-n)x) - \phi_{2,n,k}(x)| \leq \epsilon/(N^2\kappa^N).$$

Then, if we approximate $\exp(-\kappa \cdot \cos(2\pi(t-x)))$ by

$$C_0 + \sum_{n=0}^{N-1} \sum_{k\geq (n+1)/2} C_{n,k}(\cos\left(\pi(k-n)t\right)\phi_{1,n,k}(x) + \sin\left(\pi(k-n)t\right)\phi_{2,n,k}(x)),$$

the error can be bounded by

$$\epsilon + \sum_{n=0}^{N-1} \sum_{k\geq (n+1)/2} C_{n,k} \cdot \frac{2\epsilon}{N^2\kappa^N} \leq \epsilon + N^2\kappa^N \cdot \frac{2\epsilon}{N^2\kappa^N} \leq 3\epsilon,$$

which completes the proof. □

H. Proof of Theorem F.1

Given a smoothness function $\gamma\colon \mathbb{N}_0^{d\times\infty} \to \mathbb{R}$, we define

$$I(T,\gamma) \coloneqq \{(i,j) \mid \exists s \in \mathbb{N}_0^{d\times\infty} \text{ such that } s_{ij} \neq 0, \gamma(s) < T\},$$
$$d_{\max} \coloneqq |I(T,\gamma)|.$$

The *feature extraction map* $\Gamma\colon \mathbb{R}^{d\times\infty} \to \mathbb{R}^{d_{\max}}$ is defined as

$$\Gamma(X) = [X_{i_1,j_1}, \ldots, X_{i_{d_{\max}},j_{d_{\max}}}].$$

The following lemma shows that, if FNN receives finite number of "important" features, it can approximate γ-smooth functions and piecewise γ-smooth functions. This is mainly due to the condition $\|a\|_{wl^\alpha} \leq 1$, which induces sparsity of important features.

Lemma H.1 (Theorem D.3 in Takakura and Suzuki (2023))**.** *Suppose that the target functions $f \in \mathcal{F}^\gamma_{p,\theta}$ and $g \in \mathcal{P}^\gamma_{p,\theta}$ satisfy $\|f\|_\infty \leq R$ and $\|g\|_\infty \leq R$, where $R > 0$ and γ is the mixed or anisotropic smoothness and the smoothness parameter a satisfies $\|a\|_{wl^\alpha} \leq 1$. For any $T > 0$, there exist FNNs $\hat{f}_T, \hat{g}_T \in \Psi(L, W, S, B)$ such that*

$$\left\|\hat{f}_T \circ \Gamma - f\right\|_{2,P_X} \lesssim 2^{-T},$$
$$\|\hat{g}_T \circ \Gamma \circ \Pi - g\|_{2,P_X} \lesssim 2^{-T},$$

where

$$L \sim \max\left\{T^{2/\alpha}, T^2\right\}, W \sim T^{1/\alpha}2^{T/a^\dagger},$$
$$S \sim T^{2/\alpha}\max\left\{T^{2/\alpha}, T^2\right\}2^{T/a^\dagger}, \log B \sim T^{1/\alpha}.$$

From this lemma, we can see that, if the the convolution layer can approximate Γ, the SSM can give important features to the FNN, and the FNN can approximate the target function.

Now, we prove Theorem F.1.

Proof of Theorem F.1. Firstly, we construct the embedding layer $\mathrm{Emb}\colon \mathbb{R}^{d\times\infty} \to \mathbb{R}^{D\times\infty}$. Set the embedding dimension D as $\max\{d, d_{\max}\}+1$. We set $E_1 \in \mathbb{R}^{D\times d}$ to satisfy

$$E_1 x = [x_1,\ \ldots,\ x_d,\ 0,\ \underbrace{0,\ \ldots,\ 0}_{D-d-1 \text{ elements}}\ ]^\top.$$

for $x = [x_1, \ldots, x_d] \in \mathbb{R}^d$. Additionally, we set $E_2 \in \mathbb{R}^D$ to satisfy

$$E_2 = [\underbrace{0,\ \ldots,\ 0}_{d \text{ elements}},\ 1,\ \underbrace{0,\ \ldots,\ 0}_{D-d-1 \text{ elements}}\ ]^\top.$$

Note that $\|E_1\|_\infty = \|E_2\|_\infty = 1$. Then, the constructed embedding layer Emb is represented as follows:

$$\mathrm{Emb}(X) = \begin{bmatrix} \cdots & x_t & \cdots \\ \cdots & 1 & \cdots \\ \cdots & 0 & \cdots \\ \vdots & \vdots & \vdots \\ \cdots & 0 & \cdots \end{bmatrix} \in \mathbb{R}^{D\times\infty}.$$

Secondly, we construct the gated convolution layer. The role of this layer is to approximate the feature extractor Γ. The weight matrix $W^V \in \mathbb{R}^{D\times|X|}$ is set to extract the important "dimensions" $(i_1, \ldots, i_{d_{\max}})$. More precisely, we set W_V to satisfy

$$W^V y = [y_{i_1}, \ldots, y_{i_{d_{\max}}},\ \underbrace{0,\ \ldots,\ 0}_{D-d_{\max} \text{ elements}}\ ] \in \mathbb{R}^D$$

for $y = [y_1, \ldots, y_D] \in \mathbb{R}^D$. Then, the resulted projection is represented as follows:

$$W^V(\mathrm{Emb}(X)) = \begin{bmatrix} \cdots & X_{t,i_1} & \cdots \\ \vdots & \vdots & \vdots \\ \cdots & X_{t,i_{d_{\max}}} & \cdots \\ \cdots & 0 & \cdots \\ \vdots & \vdots & \cdots \\ \cdots & 0 & \cdots \end{bmatrix} \in \mathbb{R}^{D\times\infty}.$$

Next, we construct the convolution filter. From the assumption $a_{ij} = \Omega(\log(|j|+1))$, we can choose the window size $U \in \mathbb{N}$ such that

$$\log U \sim T \quad \text{and} \quad a_{ij} \le T \Longrightarrow j \le U.$$

Lemma G.4 shows that, for each j_m $(m = 1, \ldots, d_{\max})$, for any $\epsilon > 0, \kappa > 0$, there exists $k_m \in \Psi'(W', B)$ with

$$W' \lesssim \log^2 \epsilon^{-1}, \quad B \lesssim \log \epsilon^{-1} \log \kappa$$

such that

$$\max_{j=0,\ldots,U} \left| k_m\left(\frac{j}{U}\right) - \exp\left(-\kappa \cdot \sin^2\left(\frac{\pi}{2}\left(\frac{j}{U} - \frac{j_m}{U}\right)\right)\right)\right| \lesssim \epsilon.$$

Now, if $|j - j_m| \ge 1$, it holds

$$\exp\left(-\kappa \cdot \sin^2\left(\frac{\pi}{2}\left(\frac{j}{U} - \frac{j_m}{U}\right)\right)\right) \le \exp\left(-\kappa \cdot \left(\frac{2}{\pi} \cdot \frac{\pi}{2} \cdot \frac{1}{U}\right)^2\right) = \exp\left(-\frac{\kappa}{U^2}\right),$$

and, if $j = j_m$, it holds

$$\exp\left(-\kappa \cdot \sin^2\left(\frac{\pi}{2}\left(\frac{j}{U} - \frac{j_m}{U}\right)\right)\right) = 1.$$

Therefore, if we set $\kappa = U^2 \log \epsilon^{-1}$, we have

$$\max_{j=0,\dots,U} \left| k_m\left(\frac{j}{U}\right) - \delta_{j_m}(j) \right| \lesssim 2\epsilon,$$

where $\delta_{j'}$ is the function defined by

$$\delta_{j'}(j) = \begin{cases} 1 & \text{if } j = j', \\ 0 & \text{otherwise.} \end{cases}$$

This inequality show that the filter k can approximately extract the important tokens.

Finally, we set the weight matrix W^Q by

$$W^Q_{i,j} = \begin{cases} 1 & \text{if } j = d+1 \\ 0 & \text{otherwise,} \end{cases}$$

which results in $W^Q(\mathrm{Emb}(X)) = [1, \dots, 1]^\top$ and

$$\begin{aligned} g_1 \circ \mathrm{Emb}(X) &= W^Q(\mathrm{Emb}(X)) \odot (\beta^{(1)} * W^{(0)}(\mathrm{Emb}(X))) \\ &= \beta^{(1)} * W^{(0)}(\mathrm{Emb}(X)) \\ &= [z_t]_{t=-\infty}^{\infty} \in \mathbb{R}^{D \times \infty}, \\ z_t &= \sum_{s=0}^{U-1} k(s) \sum_{i=1}^{D} W^{(0)}_{i,t-s} X_{i,t-s} \\ &= \begin{bmatrix} \sum_{s=0}^{U-1} (k(s))_1 X_{i_1,t-s} \\ \vdots \\ \sum_{s=0}^{U-1} (k(s))_{d_{\max}} X_{i_{d_{\max}},t-s} \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \end{aligned}$$

Thirdly, we construct the FNN layer. From Lemma H.1, there exists an FNN $\hat{f} \in \Psi(L, W, S, B)$ such that

$$\left\| \hat{f} \circ \Gamma - F_0^\circ \right\|_{2, P_X} \lesssim 2^{-T}, \tag{H.1}$$

where

$$\begin{aligned} &L \sim \max\left\{T^{2/\alpha}, T^2\right\}, W \sim T^{1/\alpha} 2^{T/a^\dagger}, \\ &S \sim T^{2/\alpha} \max\left\{T^{2/\alpha}, T^2\right\} 2^{T/a^\dagger}, \log B \sim T^{1/\alpha}. \end{aligned} \tag{H.2}$$

Let $C \colon \mathbb{R}^D \to \mathbb{R}^d$ be a linear map such that

$$Cy = [y_1, \dots, y_{d_{\max}}]^\top$$

for $y = [y_1, \dots, y_D]^\top \in \mathbb{R}^D$, and we set $f_1 \coloneqq \hat{f} \circ C$. Note that $f_1 \in \Psi(L, W, S, B)$ for L, W, S, B defined in (H.2). The constructed data-controlled SSM $\hat{F}_t$ ($t \in \mathbb{Z}$) is represented as follows:

$$\hat{F}_t(X) = f_1(z_t) = \hat{f} \circ C(z_t) = \hat{f} \circ \hat{\Gamma} \circ \Sigma_t(X),$$

where

$$\hat{\Gamma}(X) = \left[\sum_{s=0}^{U-1} (k(s))_m X_{i_m,-s} \right]_{m=1}^{d_{\max}} \in \mathbb{R}^{d_{\max}}.$$

Now, we evaluate the error between the target function F_t° and the constructed model $\hat{F}_t$ for $t \in \mathbb{Z}$. Due to the shift-equivariance of F°, we have $F_t^\circ = F_0^\circ \circ \Sigma_t$. Additionally, we can easily check that $\hat{F}_t$ is also shift-equivariant, i.e., $\hat{F}_t(X) = \hat{F}_0(X \circ \Sigma_t)$. Moreover, since P_X is also shift-equivariant, we have $\|u\|_{2,P_X} = \|u \circ \Sigma_t\|_{2,P_X}$ for any $t \in \mathbb{Z}$ and $u \colon \mathbb{R}^{d\times\infty} \to \mathbb{R}^d$ such that $\|u\|_{2,P_X} < \infty$. Therefore, it holds

$$\left\|\hat{F}_t - F_t^\circ\right\|_{2,P_X} = \left\|\hat{F}_0 \circ \Sigma_t - F_0^\circ \circ \Sigma_t\right\|_{2,P_X} = \left\|\hat{F}_0 - F_0^\circ\right\|_{2,P_X}$$

for any $t \in \mathbb{Z}$. Therefore, it is sufficient to evaluate $\left\|\hat{F}_0 - F_0^\circ\right\|_{2,P_X}$. We evaluate the error by separating into two terms:

$$\left\|\hat{F}_0 - F_0^\circ\right\|_{2,P_X} \le \left\|\hat{F}_0 - \hat{f} \circ \Gamma\right\|_{2,P_X} + \left\|\hat{f} \circ \Gamma - F_0^\circ\right\|_{2,P_X}.$$

The second term can be bounded by (H.1), so we evaluate the first term. Since $\hat{f} \in \Psi(L, W, S, B)$ is $(BW)^L$-lipschitz continuous, for any $X \in [0,1]^{d\times\infty}$, we have

$$\left|\hat{F}_0(X) - \hat{f} \circ \Gamma(X)\right| = \left|\hat{f}(\hat{\Gamma}(X)) - \hat{f}(\Gamma(X))\right| \le (BW)^L \left\|\hat{\Gamma}(X) - \Gamma(X)\right\|_\infty.$$

Since $X \in [0,1]^{d\times\infty}$, it holds

$$\begin{aligned}
\left\|\hat{\Gamma}(X) - \Gamma(X)\right\|_\infty &= \max_{m=1,\dots,d_{\max}} \left|\sum_{s=0}^{U-1} (k(s))_m X_{i_m,-s} - \delta_{j_m}(s) X_{i_m,-s}\right| \\
&\le \max_{m=1,\dots,d_{\max}} \sum_{s=0}^{U-1} |(k(s))_m - \delta_{j_m}(s)| \\
&\le U\epsilon.
\end{aligned}$$

By setting $\epsilon = 2^{-T}/U$, we have

$$\left|\hat{F}_0(X) - \hat{f} \circ \Gamma(X)\right| \le \left\|\hat{\Gamma}(X) - \Gamma(X)\right\|_\infty \le 2^{-T}$$

for any $X \in [0,1]^{d\times\infty}$. Therefore, it holds

$$\begin{aligned}
\left\|\hat{F}_0 - F_0^\circ\right\|_{2,P_X} &\le \left\|\hat{F}_0 - \hat{f} \circ \Gamma\right\|_{2,P_X} + \left\|\hat{f} \circ \Gamma - F_0^\circ\right\|_{2,P_X} \\
&\le \sup_{X\in[0,1]^{d\times\infty}} \left|\hat{F}_0(X) - \hat{f} \circ \Gamma(X)\right| + \left\|\hat{f} \circ \Gamma - F_0^\circ\right\|_{2,P_X} \\
&\lesssim 2^{-T}.
\end{aligned}$$

Finally, we evaluate the parameters L, W, S, B which controls the class of $k \in \Psi'(W', B)$. Since $\|a\|_{wl^\alpha} = \sup_j j^\alpha \bar{a}_j^{-1} \le 1$, it holds

$$d_{\max} \coloneqq \left|\left\{(i,j) \;\middle|\; \exists s \in \mathbb{N}_0^{d\times\infty}, s_{ij} \ne 0, \gamma(s) < T\right\}\right| \le T^{1/\alpha}.$$

Therefore, we have

$$\begin{aligned}
W' &= d_{\max} \cdot \log^2 \epsilon^{-1} \lesssim T^{2+1/\alpha}, \\
\log B &\sim \log \epsilon^{-1} \log\left(U^2 \log \epsilon^{-1}\right) \lesssim T^2.
\end{aligned}$$

This completes the proof. □

I. Proof of Theorem F.2

I.1. Proof for the case of (i) γ-smooth importance function

Proof of Theorem F.2 for the case of (i) γ-smooth importance function. For $T > 0$, we define

$$I_j(T,\gamma) \coloneqq \{i \mid (i,j) \in I(T,\gamma)\} = \left\{i_1^{(j)}, \dots, i_{|I_j|}^{(j)}\right\},$$
$$r_{\max}(T,\gamma) \coloneqq \max\{j \in [J] \mid I_j(T,\gamma) \neq \emptyset\},$$

Note that $r_{\max}(T,\gamma) \sim T^{1/\alpha}$ since $a_{ij} = \Omega(j^\alpha)$.

Theorem F.1 implies that there exist an embedding layer Emb, an FNN $f_1 \in \Psi(L,W,S,B)$ and a gated convolution layer $g_1 \in \mathcal{C}(U,D,L',W',S,B)$ with

$$\begin{aligned}
&M = 1, \log U \sim T, D \sim T^{1/\alpha},\\
&L \sim T, W_1 \sim T^{1/\alpha},\\
&L' \sim \max\left\{T^{2/\alpha}, T^2\right\}, W' \sim T^{1/\alpha} 2^{T/a^\dagger},\\
&S \sim T^{2/\alpha} \max\left\{T^{2/\alpha}, T^2\right\} 2^{T/a^\dagger}, \log B \sim T^{1/\alpha},
\end{aligned}$$

such that

$$f_1 \circ g_1 \circ \mathrm{Emb}(X)_i = [x_i^\top, \widehat{\mu}_i(X), \underbrace{0, \dots, 0}_{d_{\max} \text{ elements}}, \underbrace{-1, \dots, -1}_{r_{\max} \text{ elements}}]^\top,$$

for all $i \in \mathbb{Z}$, where $\widehat{\mu}_i(X)$ satisfies

$$|\widehat{\mu}_i(X)_{-t} - (\mu_i(X) - 1)| \lesssim 2^{-T}.$$

Intuitively, the i-th elements for $i = 3, \dots, 2 + d_{\max}$ are used to store the feature $X_{t-i,j}$ for $j \in [d]$, and the i-th elements for $i = 3 + d_{\max}, \dots, 2 + d_{\max} + r_{\max}$ are buffers to store which elements are already selected. Note that, for any $i \leq r_{\max}$, it holds

$$\widehat{\mu}(X)_{\pi_\lambda(i)} - \widehat{\mu}(X)_{\pi_\lambda(i+1)} \gtrsim (\mu(X)_{\pi_\lambda(i)} - 2^{-T}) - (\mu(X)_{\pi_\lambda(i+1)} + 2^{-T}) \gtrsim T^{-\beta/\alpha},$$

and $\widehat{\mu}(X)_t \in [-1, 0]$ for all $t \in [0 : V]$.

In the following, we set $U = V$. Let us set $\chi_T \sim \dfrac{T \log 2 + 2 \log U}{T^{-\beta/\alpha}}$. Using Lemma G.2, we see that, there exists a neural network $\phi_{\exp} \in \Psi(L,W,S,B)$ with

$$L \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad W \lesssim T^{1+\beta/\alpha} \log U, \quad S \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad \log B \lesssim T^{2(1+\beta/\alpha)} \log^2 U,$$

such that, for any $x \leq 0$, it holds

$$|\phi_{\exp}(\chi_T x) - \exp(\chi_T x)| \leq 2^{-2T^{1+\beta/\alpha}}/U^3.$$

Moreover, using Lemma G.2 again, we see that there exists a neural network $\phi_\times$ with

$$L \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad W \lesssim 1, \quad S \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad \log B \lesssim T^{1+\beta/\alpha} \log U,$$

such that, for any $0 \leq x \lesssim U^2 \exp\left(T^{1+\beta/\alpha}\right)$, $0 \leq y \lesssim 1$, it holds

$$|\phi_\times(x,y) - xy| \leq 2^{-2T^{1+\beta/\alpha}}/U^3.$$

Then, for any $x \leq 0$ and $y \in [0,1]$, it holds

$$\begin{aligned}
|\phi_\times(\phi_{\exp}(\chi_T x), y) - \exp(\chi_T x) y| &\leq |\phi_\times(\phi_{\exp}(\chi_T x), y) - \phi_{\exp}(\chi_T x) y| + |\phi_{\exp}(\chi_T x) y - \exp(\chi_T x) y|\\
&\leq 2^{-2T^{1+\beta/\alpha}}/U^3 + 2^{-2T^{1+\beta/\alpha}}/U^3\\
&\lesssim 2^{-2T^{1+\beta/\alpha}}/U^3.
\end{aligned}$$

Then, let us define f_1' be an FNN layer such that it holds

$$f_1' \circ f_1 \circ g_1(X) \circ \mathrm{Emb}(X)_i = [\phi_\times(\phi_{\exp}(\widehat{\mu}_i(X)), x_i), \underbrace{0, \dots, 0}_{d_{\max} \text{ elements}}, \underbrace{-1, \dots, -1}_{r_{\max} \text{ elements}}]^\top.$$

Additionally, we define

$$Z_m \coloneqq (f_m' \circ g_m \circ f_m) \circ \cdots \circ (f_1' \circ g_1 \circ f_1) \circ \mathrm{Emb}(X),$$

for $m \in [1 : r_{\max}]$. We construct remaining layers $f_2, g_2, f_2', \dots, f_{r_{\max}+1}, g_{r_{\max}+1}, f_{r_{\max}+1}'$ to make them satisfying

$$Z_m = [\phi_\times(\phi_{\exp}(\widehat{\mu}_i(X)), x_i), \widehat{X}_{i_1^{(1)}, j_1}, \dots, \widehat{X}_{i_{|I_1|}^{(1)}, j_1}, \dots, \widehat{X}_{i_1^{(m)}, j_m}, \dots, \widehat{X}_{i_{|I_m|}^{(m)}, j_m}, \underbrace{0, \dots, 0}_{d_{\max} - \sum_{j=1}^m |I_j| \text{ elements}},$$
$$\widehat{j}_1/U, \dots, \widehat{j}_m/U, \underbrace{-1, \dots, -1}_{r_{\max} - m \text{ elements}}]^\top,$$

where $\widehat{X}_{i_k^{(j_m)}, j_m}, \widehat{j}_m$ are the approximation of $\widehat{X}_{i_k^{(j_m)}, j_m}, \widehat{j}_m$ $(m = 1, \dots, M; k = 1, \dots, |I_{j_m}|)$ respectively such that

$$\left| \widehat{X}_{i_k^{(j_m)}, j_m} - X_{i_k^{(j_m)}, j_m} \right| \lesssim 2^{-T}, \quad \left| \widehat{j}_m/U - j_m/U \right| \lesssim 2^{-3T^{1+\beta/\alpha}}/V^5.$$

Then, we see that

$$Z_M = [x_i^\top, \widehat{\mu}_i(X), \widehat{X}_{i_1^{(1)}, j_1}, \dots, \widehat{X}_{i_{|I_1|}^{(1)}, j_1}, \dots, \widehat{X}_{i_1^{(r_{\max})}, j_{r_{\max}}}, \dots, \widehat{X}_{i_{|I_{r_{\max}}|}^{(r_{\max})}, j_{r_{\max}}}, \widehat{j}_1/U, \dots, \widehat{j}_M/U]^\top.$$

Hence, Lemma H.1 shows that there exists a FNN $f_M' \in \Psi(L, W, S, B)$ with

$$L \lesssim \max\left\{T^{2/\alpha}, T^2\right\}, \quad W \lesssim T^{1/\alpha} 2^{T/\alpha^\dagger},$$
$$S \lesssim T^{2/\alpha} \max\left\{T^{2/\alpha, T^2}\right\} 2^{T/\alpha^\dagger}, \quad \log B \lesssim T^{1/\alpha},$$

such that

$$\|f_M'(Z_M) - f\|_2 \lesssim 2^{-T}.$$

The same discussion as Theorem F.1 gives the desired result.

In the following, we construct an FNN f_m and a gated convolution layer g_m for $m \in [1 : r_{\max}]$. The proof mainly divided into two parts: (i) obtaining $\widehat{X}_{i_k^{(m)}, j_m}$, i.e., the approximation of important features $X_{i_k^{(m)}, j_m}$ $(k = 1, \dots, |I_m|)$ and (ii) getting $\widehat{j}_m$, i.e., recording which token j_m was selected.

Picking up the important features $X_{i_k^{(m)}, j_m}$ $(k = 1, \dots, |I_m|)$ Due to Lemma G.1 and the fact that $j_m \in [0 : V]$ is an index such that μ_{t-j} $(\widehat{\mu}_{t-j})$ is the largest in μ_{t-j} $(\widehat{\mu}_{t-j})$ $(j \neq j_1, \dots, j_{m-1})$, for any $t \in [0 : U]$ with $t \neq t_0$, it holds

$$\left| \frac{\sum_{j=0}^V X_{i,j} \exp(\chi_T \cdot \widehat{\mu}_{t-j}) \cdot (1 - \mathbb{I}_S(j))}{\sum_{j=0}^V \exp(\chi_T \cdot \widehat{\mu}_{t-j}) \cdot (1 - \mathbb{I}_S(j))} - X_{i,j_m} \right| \leq 2U^2 \exp\left(-\chi_T \cdot T^{-\beta/\alpha}\right) \lesssim 2^{-T},$$

where $S = \{j_1, \dots, j_{m-1}\}$. Now, let us approximate

$$\frac{\sum_{j=0}^V X_{i,j} \exp(\chi_T \cdot \widehat{\mu}_{t_i}) \cdot (1 - \mathbb{I}_S(j))}{\sum_{j=0}^V \exp(\chi_T \cdot \widehat{\mu}_{t-i}) \cdot (1 - \mathbb{I}_S(j))} = \frac{\frac{1}{V}\sum_{j=0}^V X_{i,j} \exp(\chi_T \cdot \widehat{\mu}_{t_i}) \cdot (1 - \mathbb{I}_S(j))}{\frac{1}{V}\sum_{j=0}^V \exp(\chi_T \cdot \widehat{\mu}_{t-i}) \cdot (1 - \mathbb{I}_S(j))}$$

using neural networks. Using Lemma G.2, we see that, for any $\epsilon_1 > 0$, there exists a neural network $\phi_{\mathrm{rec}} \in \Psi(L, W, S, B)$ with

$$L \lesssim \log^2 \epsilon_1^{-1}, \quad W \lesssim \log^3 \epsilon_1^{-1}, \quad S \lesssim \log^4 \epsilon_1^{-1}, \quad \log B \lesssim \log \epsilon_1^{-1},$$

such that, for any $x \in [\epsilon_1, \epsilon_1^{-1}]$, it holds

$$\left| \phi_{\mathrm{rec}}(x) - \frac{1}{x} \right| \leq \epsilon_1,$$

and $\phi_{\mathrm{rec}}(x) \in (0,1]$. Moreover, using Lemma G.2 again, we see that, for any $\epsilon_3 > 0$, there exists a neural network $\phi_{\mathrm{mult}} \in \Psi(L, W, S, B)$ with

$$L \lesssim \log \epsilon_2^{-1}, \quad W \lesssim 1, \quad S \lesssim \log \epsilon_2^{-1}, \quad \log B \lesssim 1,$$

such that, for any $x \in [0,1]^2$, it holds

$$|\phi_{\mathrm{mult}}(x) - x_1 x_2| \le \epsilon_2.$$

Setting $\epsilon_1 = \min\{\exp(-\chi_T), 2^{-T}\}, \epsilon_2 = 2^{-T}$ and

$$\phi_* \colon [\exp(-\chi_T), 1] \times [0,1] \to \mathbb{R}, (x,y) \mapsto \phi_{\mathrm{mult}}(\phi_{\mathrm{rec}}(x), y),$$

we see that $\phi_* \in \Psi(L, W, S, B)$ with

$$L \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad W \lesssim T^{3(1+\beta/\alpha)} \log^3 U, \quad S \lesssim T^{4(1+\beta/\alpha)} \log^4 U, \quad \log B \lesssim T^{1+\beta/\alpha} \log U,$$

and, for any $x \in [\exp(-\chi_T), \exp(\chi_T)], y \in [0, U]$, it holds

$$\begin{aligned}
\left|\phi_*(x,y) - \frac{y}{x}\right| &= \left|\phi_{\mathrm{mult}}(\phi_{\mathrm{rec}}(x), y) - \frac{y}{x}\right| \\
&\le |\phi_{\mathrm{mult}}(\phi_{\mathrm{rec}}(x), y) - y\phi_{\mathrm{rec}}(x)| + \left|y\phi_{\mathrm{rec}}(x) - \frac{y}{x}\right| \\
&\le \epsilon_2 + \epsilon_1 \lesssim 2^{-T}.
\end{aligned}$$

Now, if we set $w(x,y) \coloneqq y/x$ for $x, y > 0$, we have

$$\|\nabla w(x,y)\|_2 = \sqrt{\left(\frac{1}{x}\right)^2 + \left(-\frac{y}{x^2}\right)^2} = \frac{1}{x}\sqrt{1 + \left(\frac{y}{x}\right)^2}.$$

Therefore, if $x \ge 1$ and $0 \le y \le x$, it holds

$$\|\nabla w(x,y)\|_2 \le \frac{1}{x}\sqrt{1 + \left(\frac{x}{x}\right)^2} = \sqrt{2}\exp(\chi_T) \lesssim U^2 2^{T^{1+\beta/\alpha}},$$

which means, for any $x, x', y, y' > 0$ with $x, x' \in [\exp(-\chi_T), 1]$ and $0 \le y \le x, 0 \le y' \le x'$, it holds

$$\left|\frac{y}{x} - \frac{y'}{x'}\right| \lesssim U^2 2^{T^{1+\beta/\alpha}}(|x - x'| + |y - y'|),$$

which means

$$\begin{aligned}
\left|\phi_*(x',y') - \frac{y}{x}\right| &\le \left|\phi_1(x',y') - \frac{y'}{x'}\right| + \left|\frac{y'}{x'} - \frac{y}{x}\right| \\
&\lesssim 2^{-T} + U^2 2^{T^{1+\beta/\alpha}}(|x - x'| + |y - y'|)
\end{aligned}$$

Next, Lemma G.4 implies that there exists a neural networks $\phi_n' \in \Psi'(1, B)$ and $\phi_n \in \Psi(L, W, S, B)$ $(n = 1, \dots, N)$ with

$$\begin{aligned}
&N \lesssim T^{1+\beta/\alpha} \log T \log V, \\
&L \lesssim T^{2(1+\beta/\alpha)} \log T \log^2 V, \quad W \lesssim 1, \\
&S \lesssim T^{2(1+\beta/\alpha)} \log T \log^2 V, \quad \log B \lesssim 1,
\end{aligned}$$

such that, for any $t, x, \widehat{x} \in [0,1]$, it holds

$$\begin{aligned}
&\left|\sum_{n=0}^{N} \phi_n'(t)\phi_n(\widehat{x}) - \exp\left(-\frac{V^2(\frac{1}{\alpha}\log T + 2T^{1+\beta/\alpha} + 2\log V) \cdot \sin^2\left(\frac{\pi}{2}(t-x)\right)}{2}\right)\right| \\
&\qquad \lesssim T^{-1/\alpha} 2^{-2T^{1+\beta/\alpha}}/V^2 + T^{1+\beta/\alpha} V^3 |x - \hat{x}|.
\end{aligned}$$

Since

$$\exp\left(-\frac{V^2(\frac{1}{\alpha}\log T + 2T^{1+\beta/\alpha} + 2\log V)\sin^2\left(\frac{\pi}{2}(t-x)\right)}{2}\right) \begin{cases} \leq T^{-1/\alpha}2^{-2T^{1+\beta/\alpha}}/V^2 & (|t-x| \geq 1/V), \\ = 1 & (t=x), \end{cases}$$

we have

$$\left|\exp\left(-\frac{V^2(\frac{1}{\alpha}\log T + 2T^{1+\beta/\alpha} + 2\log V)\sin^2\left(\frac{\pi}{2}(t-x)\right)}{2}\right) - \mathbb{I}_{\{x\}}(t)\right| \lesssim 2T^{-1/\alpha}2^{-2T^{1+\beta/\alpha}}/V^2.$$

Therefore, we have

$$\left|\sum_{n=1}^{N}\phi_n'(t)\phi_n(\widehat{x}) - \mathbb{I}_{\{x\}}(t)\right| \lesssim T^{-1/\alpha}2^{-(1+\beta/\alpha)T}/V^2 + T^{1+\beta/\alpha}V^3|x-\hat{x}|.$$

Summing up over $x = j_1/U, \ldots, j_{m-1}/U$, we have

$$\begin{aligned}
&\left|\sum_{m'=1}^{m-1}\sum_{i=1}^{I}\phi_0^{(j_{m'},i)}(t)\phi_1^{(j_{m'},i)}(\hat{j}_{m'}/U) - \mathbb{I}_S(t)\right| \\
&\quad\lesssim r_{\max}\left(T^{-1/\alpha}2^{-2T^{1+\beta/\alpha}}/V^2 + T^{1+\beta/\alpha}V^3\left|\hat{j}_{m'}/U - j_{m'}/U\right|\right) \\
&\quad\lesssim 2^{-2T^{1+\beta/\alpha}}/V^2.
\end{aligned}$$

Combining the results above, we have

$$\begin{aligned}
&\left|\frac{1}{V}\sum_{j=0}^{V}\phi_\times(\phi_{\exp}(X_{i,j}, \chi_T\widehat{\cdot}\widehat{\mu}_{t-j}))\cdot\left(1 - \sum_{m'=1}^{m-1}\sum_{i=1}^{I}\phi_0^{(j_{m'},i)}(t)\phi_1^{(j_{m'},i)}(\hat{j}_{m'}/U)\right)\right. \\
&\qquad\left. - \frac{1}{V}\sum_{j=0}^{V}X_{i,j}\exp(\chi_T\cdot\widehat{\mu}_{t-j})\cdot(1-\mathbb{I}_S(j))\right| \\
&\lesssim \frac{1}{V}\sum_{j=0}^{V}\left(\left|\phi_\times(\phi_{\exp}(X_{i,j}, \chi_T\cdot\widehat{\mu}_{t-j}))\cdot\left(\sum_{m'=1}^{m-1}\sum_{i=1}^{I}\phi_0^{(j_{m'},i)}(t)\phi_1^{(j_{m'},i)}(\hat{j}_{m'}/U) - \mathbb{I}_S(j)\right)\right|\right. \\
&\qquad\left. + |(\phi_\times(\phi_{\exp}(X_{i,j}, \chi_T\widehat{\cdot}\widehat{\mu}_{t-j})) - X_{i,j}\exp(\chi_T\cdot\widehat{\mu}_{t-j}))\mathbb{I}_S(j)|\right) \\
&\lesssim 2^{-2T^{1+\beta/\alpha}}/V^2
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
&\left|\frac{1}{V}\sum_{j=0}^{V}\phi_{\exp}(\chi_T\widehat{\cdot}\widehat{\mu}_{t-j})\cdot\left(1 - \sum_{m'=1}^{m-1}\sum_{i=1}^{I}\phi_0^{(j_{m'},i)}(t)\phi_1^{(j_{m'},i)}(\hat{j}_{m'}/U)\right)\right. \\
&\qquad\left. - \frac{1}{V}\sum_{j=0}^{V}\exp(\chi_T\cdot\widehat{\mu}_{t-j})\cdot(1-\mathbb{I}_S(j))\right| \\
&\lesssim 2^{-2T^{1+\beta/\alpha}}/V^2
\end{aligned}$$

Using the facts that

$$\exp(-\chi_T) \le \frac{1}{V}\sum_{t=0}^{V}\exp(\chi_T \cdot \widehat{\mu}[t]) \le 1,$$
$$\frac{1}{V}\sum_{t=0}^{V} u[t]\exp(\chi_T \cdot \widehat{\mu}[t]) \le \frac{1}{V}\sum_{t=0}^{V}\exp(\chi_T \cdot \widehat{\mu}[t]),$$

we have

$$\begin{aligned}
&\left|\phi_1\left(\frac{1}{V}\sum_{j=0}^{V}\phi_\times(X_{i,j},\phi_{\exp}(\widehat{\chi}_T\cdot\mu_{t-j}))\cdot\left(1-\sum_{m'=1}^{m-1}\sum_{i=1}^{I}\phi_0^{(j_{m'},i)}(t)\phi_1^{(j_{m'},i)}(\hat{j}_{m'}/U)\right),\right.\right.\\
&\qquad\left.\left.\frac{1}{V}\sum_{j=0}^{V}\phi_{\exp}(\widehat{\chi}_T\cdot\mu_{t-j})\cdot\left(1-\sum_{m'=1}^{m-1}\sum_{i=1}^{I}\phi_0^{(j_{m'},i)}(t)\phi_1^{(j_{m'},i)}(\hat{j}_{m'}/U)\right)\right)\right.\\
&\qquad\left.-\frac{\sum_{t=0}^{U}u[t]\exp(\chi_T\cdot\widehat{\mu}[t])\cdot(1-\mathbb{I}_S(j))}{\sum_{t=0}^{U}\exp(\chi_T\cdot\widehat{\mu}[t])\cdot(1-\mathbb{I}_S(j))}\right|\\
&\lesssim 2^{-T}+V^2 2^{T^{1+\beta/\alpha}}\cdot 2^{-2T^{1+\beta/\alpha}}/V^2 \lesssim 2^{-T}.
\end{aligned}$$

Overall, we can see that, there exist neural networks $\phi_O \in \Psi'(L,W,S,B)$ and $\phi_A,\phi_B,\phi_C \in \Psi(L,W,S,B)$ with

$$\begin{aligned}
&L \lesssim T^{3+1/\alpha+3\beta/\alpha}\log T\log^3 V, \quad W \lesssim T^{2+1/\alpha+2\beta/\alpha}\log T\log^2 V,\\
&S \lesssim T^{3+1/\alpha+3\beta/\alpha}\log T\log^3 V, \quad \log B \lesssim T^{3+1/\alpha+3\beta/\alpha}\log T\log^3 V,
\end{aligned}$$

such that

$$\max_{i\in\left\{i_1^{(m)},\ldots,i_{|I_m|}^{(m)}\right\}}\Bigg|\underbrace{\phi_C\left(\sum_{j=0}^{V}\phi_O(j/V)\phi_A(Z_{m-1})\phi_B(Z_{m-1}[-j])\right)}_{=:\widehat{X}_{i,j_m}}-X_{j_m,i}\Bigg| \lesssim 2^{-T}.$$

Recording which token was picked up Similar discussion as above shows that there exist neural networks $\phi'_O \in \Psi'(L,W,S,B)$ and $\phi'_A,\phi'_B,\phi'_C \in \Psi(L,W,S,B)$ with

$$\begin{aligned}
&L \lesssim T^{3+1/\alpha+3\beta/\alpha}\log T\log^3 V, \quad W \lesssim T^{2+1/\alpha+2\beta/\alpha}\log T\log^2 V,\\
&S \lesssim T^{3+1/\alpha+3\beta/\alpha}\log T\log^3 V, \quad \log B \lesssim T^{3+1/\alpha+3\beta/\alpha}\log T\log^3 V,
\end{aligned}$$

such that

$$\left|\phi'_C\left(\sum_{j=0}^{V}\phi'_O(j/V)\phi'_A(Z_{m-1})\phi'_B(Z_{m-1}[-j])\right)-\sin\left(\frac{\pi}{4}\frac{j_m}{V}\right)\right| \lesssim 2^{-T}.$$

Lemma G.2 shows that there exists a neural network $\phi_{\arcsin} \in \Psi(L,W,S,B)$ with

$$L \lesssim T^{2(1+\beta/\alpha)}\log^2 V, \quad W \lesssim 1, \quad S \lesssim T^{2(1+\beta/\alpha)}\log^2 V, \quad \log B \lesssim T^{1+\beta/\alpha}\log V,$$

for any $x \in [0,\pi/4]$, it holds

$$|\phi_{\arcsin}(x)-\arcsin(x)| \lesssim 2^{-3T^{1+\beta/\alpha}}/V^5.$$

Using this network, we can obtain $\widehat{j}_m/V$ such that $\left|\widehat{j}_m/V - j_m/V\right| \lesssim 2^{-3T^{1+\beta/\alpha}}/V^5$.

Finishing the proof We can easily see that, constructing the weight matrix in the gated convolution layers appropriately, we can obtain Z_m from Z_{m-1} using the neural networks constructed above. This completes the proof. □

I.2. Proof for the case of (ii) importance functions with similarity

In this subsection, we consider the case of similarity-based importance function.

First, we show the approximation error when the importance is given by the distance. Thanks to the separated condition of the importance function, we can see that, for any $j \neq j'$, it holds

$$\|v_0 - v_j\|^2 - \|v_0 - v_{j'}\|^2 = (\|v_0 - v_j\| - \|v_0 - v_{j'}\|)(\|v_0 + v_j\| + \|v_0 + v_{j'}\|) \gtrsim T^{-2\beta/\alpha}.$$

Now, since it is hold that

$$u^2 - \frac{u^4}{3} \leq \sin^2(u) = \frac{1 - \cos 2u}{2} \leq u^2,$$

for $u \in [0, \pi/2]$, for $A > 0$, it holds

$$\begin{aligned}
&\left| \left(A \sin\left(\frac{\pi}{2A}(v_{0i} - v_{ji}) \right) \right)^2 - \left(\frac{\pi}{2}(v_{0i} - v_{ji}) \right)^2 \right| \\
&\qquad = \left| A^2 \sin^2\left(\frac{\pi}{2A}(v_{0i} - v_{ji}) \right) - A^2 \left(\frac{\pi}{2A}(v_{0i} - v_{ji}) \right)^2 \right| \\
&\qquad \lesssim \frac{1}{A^2}.
\end{aligned}$$

Therefore, if we set $A \sim \sqrt{d'} T^{\beta/\alpha}$, it holds

$$\sum_{i=1}^{d'} \left(A \sin\left(\frac{\pi}{2A}(v_{0i} - v_{ji}) \right) \right)^2 - \sum_{i=1}^{d'} \left(A \sin\left(\frac{\pi}{2A}(v_{0i} - v_{ji'}) \right) \right)^2 \gtrsim T^{-2\beta/\alpha}.$$

Then, let us set $\kappa \sim \dfrac{T + \log V}{T^{-2\beta/\alpha}}$. Therefore, if we can approximate

$$\exp\left(-\kappa \sum_{i=1}^{d'} \left(A \sin\left(\frac{\pi}{2A}(v_{0i} - v_{ji}) \right) \right)^2 \right) = \prod_{i=1}^{d'} \exp\left(-\kappa A^2 \sin^2\left(\frac{\pi}{2}(v_{0i} - v_{ji}) \right) \right),$$

with the error less then $2^{-2T^{1+\beta/\alpha}}/V^3$ efficiently, then the same discussion as the case of (i) gives the desired result, due to Lemma G.1.

Using Lemma G.4, we can see that there exists sneural network $\phi_n^{(i)}, \psi_n^{(i)} \in \Psi(L, W, S, B)$ $(n = 1, \ldots, N)$ with

$$\begin{aligned}
&N \lesssim T^{1+\beta/\alpha} \log V, \\
&L \lesssim T^{2(1+\beta/\alpha)} \log^2 T \log^3 V, \quad W \lesssim T^{2(1+\beta/\alpha)} \log^2 V, \\
&S \lesssim T^{4(1+\beta/\alpha)} \log^2 T \log^5 V, \quad \log B \lesssim T^{1+\beta/\alpha} \log T \log^2 V,
\end{aligned}$$

such that, for any $x, y \in [0, 1]$, it holds

$$\left| \sum_{n=1}^{N} \phi_n^{(i)}(x) \psi_n^{(i)}(y) - \exp\left(-\kappa \sin^2\left(\frac{\pi}{2}(x - y) \right) \right) \right| \lesssim \frac{2^{-2T^{1+\beta/\alpha}}}{{d'}^2 V^3}.$$

Since we can see that $\exp\bigl(-\kappa \sin^2(\pi/2(v_{0i} - vji))\bigr) \in (0, 1]$, it holds

$$\left| \prod_{i=1}^{d'} \left(\sum_{n=1}^{N} \phi_n^{(i)}(v_{0i}) \psi_n^{(i)}(v_{ij}) \right) - \prod_{i=1}^{d'} \exp\left(-\kappa A^2 \sin^2\left(\frac{\pi}{2}(v_{0i} - v_{ji}) \right) \right) \right| \lesssim \frac{2^{-2T^{1+\beta/\alpha}}}{d' V^3}.$$

Finally, Lemma G.2 shows that there exists a neural network $\phi_\times$ with

$$L \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad W \lesssim 1, \quad S \lesssim T^{2(1+\beta/\alpha)} \log^2 U, \quad \log B \lesssim T^{1+\beta/\alpha} \log U,$$

such that

$$\left|\phi_{\times}\left(\left[\sum_{n=1}^{N}\phi_n^{(i)}(v_{0,i})\psi_n^{(i)}(v_{j,i})\right]_{i=1}^{d'}\right)-\prod_{i=1}^{d'}\left(\sum_{n=1}^{N}\phi_n^{(i)}(v_{0,i})\psi_n^{(i)}(v_{j,i})\right)\right|\lesssim 2^{-2T^{1+\beta/\alpha}}/V^3,$$

which completes the proof.

As for the setting of inner product, we have

$$\exp\left(v_0^\top v_j\right)=\exp\left(\frac{1}{2}\|v_0\|^2\right)\exp\left(\frac{1}{2}\|v_j\|^2\right)\exp\left(-\frac{1}{2}\|v_0-v_j\|^2\right).$$

Since $\exp\left(\frac{1}{2}\|v_0\|^2\right)$ and $\exp\left(\frac{1}{2}\|v_j\|^2\right)$ can be approximated by neural networks in each token, we immediately obtain the desired result.

J. Proof of Theorem 4.2 and Theorem 4.4

To establish the theory on the estimation ability of SSMs, we first introduce the following theorem, which evaluates the estimation ability of ERM estimators in $\mathcal{S}(M,U,D,L,W,S,B)$.

Theorem J.1. *Let $\hat{F}\in\mathcal{S}(M,U,D,L,W,S,B)$ be an ERM estimator which minimizes the emprical cost. Then, for any $\delta\in(0,1)$, it holds that*

$$R_{l,r}(\hat{F},F^\circ)\lesssim\inf_{F\in\mathcal{S}}\frac{1}{r-l+1}\sum_{i=l}^{r}\|F_i-F_i^\circ\|_{2,P_X}^2+\frac{1}{n}\cdot M^2L(S+D)\log\left(\frac{DULWB}{\delta}\right)+\delta.$$

This theorem can be proved by using Theorem 5.2 of Takakura and Suzuki (2023) and the bound of covering number of the space $\mathcal{S}$. The proof can be found in Appendix J.

To prove the theorem, we use the following proposition.

Proposition J.2 (Theorem 5.2 in Takakura and Suzuki (2023))**.** *For a given class $\mathcal{F}$ of functions from $[0,1]^{d\times\infty}$ to $\mathbb{R}^\infty$, let $\hat{F}\in\mathcal{F}$ be an ERM estimator which minimizes the empirical cost. Suppose that there exists a constant $R>0$ such that $\|F^\circ\|_\infty\le R$, $\|F\|_\infty\le R$ for any $F\in\mathcal{F}$, and $\mathcal{N}(\mathcal{F},\delta,\|\cdot\|_\infty)\ge 3$. Then, for any $0<\delta<1$, it holds that*

$$R_{l,r}(\hat{F},F^\circ)\lesssim\inf_{F\in\mathcal{F}}\frac{1}{r-l+1}\sum_{i=l}^{r}\|F_i-F_i^\circ\|_{2,P_X}^2+(R^2+\sigma^2)\frac{\log\mathcal{N}(\mathcal{F},\delta,\|\cdot\|_\infty)}{n}+(R+\sigma)\delta,$$

where $\mathcal{N}(\mathcal{F},\delta,\|\cdot\|)$ is the δ-covering number of the space $\mathcal{F}$ associated with the norm $\|\cdot\|$, defined by

$$\mathcal{N}(\mathcal{F},\delta,\|\cdot\|):=\inf\{m\in\mathbb{N}\mid\exists F_1,\dots,F_m\in\mathcal{F},\forall F\in\mathcal{F},\exists i\in[m]\ \textit{s.t.}\ \|F-F_i\|\le\delta\}.$$

Thanks to this proposition, the problem to obtain the upper bound of the excess risk of the estimator $\hat{F}$ is reduced to the problem to evaluate the covering number of the function class $\mathcal{S}$. The covering number of the function class $\mathcal{S}$ can be evaluated as follows.

Theorem J.3 (Covering number of SSMs with gated convolution)**.** *The covering number of the function class $\mathcal{S}(M,U,D,L,W,S,B)$ can be bounded as*

$$\log\mathcal{N}(\mathcal{S}(M,U,D,L,W,S,B),\delta,\|\cdot\|_\infty)\lesssim M^2L(S+D^2)\log\left(\frac{DULWB}{\delta}\right).$$

This theorem implies that the upper bound of the covering number of the function class $\mathcal{S}$ polynomially increases with respect to the embedding dimensions D, the number of layers M,L and the sparsity S of the parameters. This result is similar to the result by Takakura and Suzuki (2023) on the covering number of Transformers.

A large difference of the covering number between the SSMs and Transformers is the dependence on the window size U; the covering number of the SSMs depends on U logarithmically, while that of the Transformers does not depend on U. This is because SSMs sum up the tokens in the convolution without normalization. Whereas it is prefered that the covering number does not depend on U, the logarithmic dependence on U is not a serious problem for the estimation ability, as we will see later.

In the following, we prove Theorem J.3. First of all, we introduce the lemma below, which is useful to evaluate the covering number.

Lemma J.4. *Let $\{f_\theta\}_{\theta\in\Theta}$ be a parametrized function class from $[0,1]^{d\times\infty}$ to $\mathbb{R}^\infty$. Suppose that the parameter space Θ satisfies $\Theta \subseteq [-B,B]^D$ for some $B>0, D>0$. Additionally, suppose that*

$$|\{\theta \mid \theta \neq 0, \theta \in \Theta\}| \leq S.$$

Moreover, assume that there exists a constant $r>0$ such that

$$\left\|f_\theta - f_{\tilde{\theta}}\right\|_\infty \leq r\|\theta - \tilde{\theta}\|_\infty \quad \text{for any } \theta, \tilde{\theta} \in \Theta.$$

Then, it holds

$$\log \mathcal{N}(\mathcal{F}, \delta, \|\cdot\|_\infty) \leq S \log\left(\frac{rBD}{\delta}\right).$$

The following lemma is drawn from Takakura and Suzuki (2023), which evaluates the norm of the output of FNN, the lipschitz constant with respect to the input, and the lipschitz constant with respect to the parameters.

Lemma J.5 (Lemma E.3 in Suzuki (2018))**.** *Suppose that two FNNs $f, \widetilde{f}$ with L layers and W hidden units is given by*

$$f(x) \coloneqq (A_L\sigma(\cdot) + b_L) \circ \cdots \circ (A_1\sigma(x) + b_1),$$
$$\widetilde{f}(x) \coloneqq (\widetilde{A}_L\sigma(\cdot) + \widetilde{b}_L) \circ \cdots \circ (\widetilde{A}_1\sigma(x) + \widetilde{b}_1),$$

where σ is the ReLU activation function. Assume that for any $l=1,\ldots,L$, it holds

$$\|A_l\|_\infty \leq B, \quad \left\|\widetilde{A}_l\right\|_\infty \leq B, \quad \|b_l\|_\infty \leq B, \quad \left\|\widetilde{b}_l\right\|_\infty \leq B.$$

Additionally, let $r \geq 1$ be a constant.

1. *For any $x \in \mathbb{R}^{D\times\infty}$ with $\|x\|_\infty \leq r$, it holds*

$$\|f(x)\|_\infty \leq (2BW)^L r.$$

2. *For any $X, X' \in \mathbb{R}^{D\times\infty}$, it holds*

$$\|f(x) - f(x')\|_\infty \leq (BW)^L \|X - X'\|_\infty.$$

3. *Assume that, for any $l=1,\ldots,L$, it holds*

$$\left\|A_l - \widetilde{A}_l\right\|_\infty \leq \delta, \quad \left\|b_l - \widetilde{b}_l\right\|_\infty \leq \delta.$$

Then, for any $x \in \mathbb{R}^D$ with $\|x\|_\infty \leq r$, it holds

$$\left\|f(x) - \widetilde{f}(x)\right\|_\infty \leq 2(2BW)^L r \cdot \delta.$$

We also evaluate them for the gated convolution layers.

Lemma J.6. *Suppose that two gated convolution layers $g, \widetilde{g}$ with window size U and embedding dimention D is given by*[1]

$$g(X) \coloneqq (W_Q X) \odot (\beta(X) * (W_V X)),$$
$$\widetilde{g}(X) \coloneqq (\widetilde{W}_Q X) \odot (\widetilde{\beta}(X) * (\widetilde{W}_V X)).$$

Let $r \geq 1$ be a constant. Assume that it holds

$$\|W_Q\|_\infty \leq B, \quad \left\|\widetilde{W}_Q\right\|_\infty \leq B, \quad \|W_V\|_\infty \leq B, \quad \left\|\widetilde{W}_V\right\|_\infty \leq B,$$

and, for any $h = 0, \ldots, H$ and $X \in \mathbb{R}^{d \times \infty}$ with $\|X\|_\infty \leq r$, it holds

$$\|\beta(X)\|_1 \leq c, \quad \left\|\widetilde{\beta}(X)\right\|_1 \leq c,$$

for some $B \geq 1, c \geq 1$. Then, the following statements hold.

1. *For any $X \in \mathbb{R}^{D \times \infty}$ with $\|X\|_\infty \leq r$, it holds*

$$\|g(X)\|_\infty \leq (BDrc)^2.$$

2. *Suppose that $X, X' \in \mathbb{R}^{D \times \infty}$ satisfies $\|X\|_\infty \leq r, \|X'\|_\infty \leq r$ and*

$$\|\beta(X) - \beta(X')\|_1 \leq \kappa \|X - X'\|_\infty$$

for some $\kappa \geq 0$[2]*. Then, it holds*

$$\|g(X) - g(X')\|_\infty \leq \left(2B^2 rc + Br \cdot \kappa\right) \|X - X'\|_\infty.$$

3. *Assume that, for any $h = 0, \ldots, H$, it holds*

$$\left\|W_Q - \widetilde{W}_Q\right\|_\infty \leq \delta, \quad \left\|W_V - \widetilde{W}_V\right\|_\infty \leq \delta, \quad \left\|\beta(X) - \widetilde{\beta}(X)\right\|_1 \leq \iota\delta.$$

for $\iota > 0$. Then, it holds

$$\|g(X) - \widetilde{g}(X)\|_\infty \leq \left(2Br^2 c + (Br)^2 \cdot \iota\right) \cdot \delta.$$

Proof. We use frequently the following three inequalities:

$$\|WX\|_\infty \leq \|W\|_1 \|X\|_\infty \leq D \cdot \|W\|_\infty \|X\|_\infty,$$
$$\|X \odot Y\|_\infty \leq \|X\|_\infty \|Y\|_\infty,$$
$$\|\beta * X\|_\infty \leq \|\beta\|_1 \|X\|_\infty,$$

where $W \in \mathbb{R}^{D \times D}, X \in \mathbb{R}^{D \times \infty}, Y \in \mathbb{R}^{D \times \infty}, \beta \in \mathbb{R}^{D \times U}$.

Proof of 1 We have

$$\begin{aligned}
\|g(X)\|_\infty &= \|(W_Q X) \odot (\beta(X) * (W_V X))\|_\infty \\
&\leq \|W_Q X\|_\infty \cdot \|\beta(X) * (W_V X)\|_\infty \\
&\leq \|W_Q X\|_\infty \cdot \|W_V X\|_\infty \cdot \|\beta(X)\|_1 \\
&\leq (BDr)^2 \cdot c \leq (BDrc)^2.
\end{aligned}$$

[1] This architecture can be easily extended to the multi-order version since it corresponds to $g_H \circ g_{H-1} \circ \cdots \circ g_1$ with $W_V = I$ for $g_2, \ldots, g_H$.

[2] If the filter is not data-controlled, then $\kappa = 0$.

Proof of 2 We have

$$\begin{aligned}
\|g(X) - g(X')\|_\infty &= \|(W_Q X) \odot (\beta(X) * (W_V X)) - (W_Q X') \odot (\beta(X') * (W_V X'))\|_\infty \\
&\le \|(W_Q X) \odot (\beta(X) * (W_V X)) - (W_Q X') \odot (\beta(X) * (W_V X))\|_\infty \\
&\quad + \|(W_Q X') \odot (\beta(X) * (W_V X)) - (W_Q X') \odot (\beta(X') * (W_V X))\|_\infty \\
&\quad\quad + \|(W_Q X') \odot (\beta(X') * (W_V X)) - (W_Q X') \odot (\beta(X') * (W_V X'))\|_\infty \\
&\le \|(W_Q (X - X')) \odot (\beta(X) * (W_V X))\|_\infty \\
&\quad + \|(W_Q X') \odot ((\beta(X) - \beta(X')) * (W_V X))\|_\infty \\
&\quad\quad + \|(W_Q X') \odot (\beta(X') * (W_V (X - X')))\|_\infty \\
&\le \|W_Q (X - X')\|_\infty \cdot \|\beta(X)\|_1 \cdot \|W_V X\|_\infty \\
&\quad + \|W_Q X'\|_\infty \cdot \|\beta(X) - \beta(X')\|_1 \cdot \|W_V X\|_\infty \\
&\quad\quad + \|W_Q X'\|_\infty \cdot \|\beta(X')\|_1 \cdot \|W_V (X - X')\|_\infty \\
&\le B\|X - X'\|_\infty \cdot c \cdot Br + Br \cdot \kappa \|X - X'\|_\infty \cdot Br + Br \cdot c \cdot B\|X - X'\|_\infty \\
&= \big(2B^2 rc + Br \cdot \kappa\big)\|X - X'\|_\infty.
\end{aligned}$$

Proof of 3 We have

$$\begin{aligned}
\|g(X) - \widetilde{g}(X)\|_\infty &= \left\|(W_Q X) \odot (\beta(X) * (W_V X)) - (\widetilde{W}_Q X) \odot \left(\widetilde{\beta}(X) * \left(\widetilde{W}_V X\right)\right)\right\|_\infty \\
&\le \left\|(W_Q X) \odot (\beta(X) * (W_V X)) - (\widetilde{W}_Q X) \odot (\beta(X) * (W_V X))\right\|_\infty \\
&\quad + \left\|(\widetilde{W}_Q X) \odot (\beta(X) * (W_V X)) - (\widetilde{W}_Q X) \odot \left(\widetilde{\beta}(X) * (W_V X)\right)\right\|_\infty \\
&\quad\quad + \left\|(\widetilde{W}_Q X) \odot \left(\widetilde{\beta}(X) * (W_V X)\right) - (\widetilde{W}_Q X) \odot \left(\widetilde{\beta}(X) * \left(\widetilde{W}_V X\right)\right)\right\|_\infty \\
&\le \left\|\left(\left(W_Q - \widetilde{W}_Q\right)X\right) \odot (\beta(X) * (W_V X))\right\|_\infty \\
&\quad + \left\|(\widetilde{W}_Q X) \odot \left(\left(\beta(X) - \widetilde{\beta}(X)\right) * (W_V X)\right)\right\|_\infty \\
&\quad\quad + \left\|(\widetilde{W}_Q X) \odot \left(\widetilde{\beta}(X) * \left(\left(W_V - \widetilde{W}_V\right)X\right)\right)\right\| \\
&\le \left\|(W_Q - \widetilde{W}_Q)X\right\|_\infty \cdot \|\beta(X)\|_1 \cdot \|W_V X\|_\infty \\
&\quad + \left\|\widetilde{W}_Q X\right\|_\infty \cdot \left\|\beta(X) - \widetilde{\beta}(X)\right\|_1 \cdot \|W_V X\|_\infty \\
&\quad\quad + \left\|\widetilde{W}_Q X\right\|_\infty \cdot \left\|\widetilde{\beta}(X)\right\|_1 \cdot \left\|\left(W_V - \widetilde{W}_V\right)X\right\|_\infty \\
&\le \delta r \cdot c \cdot Br + Br \cdot \iota\delta \cdot Br + Br \cdot c \cdot \delta r \\
&= \big(2Br^2 c + (Br)^2 \cdot \iota\big)\delta.
\end{aligned}$$

□

Subsequently, we evaluate the lipschitz constant of the composition of the layers with respect to the input and the parameters.

Lemma J.7. *Let $(f_1, \widetilde{f}_1), \ldots, (f_M, \widetilde{f}_M)$ be pairs of two FNNs which satisfy the same condition of the pair $(f, \tilde{f})$ in Lemma J.5. Additionally, let $(g_1, \widetilde{g}_1), \ldots, (g_M, \widetilde{g}_M)$ be gated convolution layers which satisfy the same condition of the pair $(g, \widetilde{g})$ in Lemma J.6. Suppose $R > 0$ be a constant, and $F, \widetilde{F} \colon [0,1]^{d \times \infty} \to \mathbb{R}^\infty$ are two functions defined by*

$$\begin{aligned}
F &\coloneqq \mathrm{clip}_R \circ f_M \circ g_M \circ \cdots \circ \mathrm{clip}_R \circ f_1 \circ g_1, \\
\widetilde{F} &\coloneqq \mathrm{clip}_R \circ \widetilde{f}_M \circ \widetilde{g}_M \circ \cdots \circ \mathrm{clip}_R \circ \widetilde{f}_1 \circ \widetilde{g}_1.
\end{aligned}$$

Moreover, assume that $B \ge 1, c \ge 1, r \ge 1$. Then, it holds

$$\left\|F(X) - \widetilde{F}(X)\right\|_\infty \le 2^{M+1}(2BW)^{ML}(BDRc)^{2M}(1+\kappa)^M(1+\iota) \cdot \delta.$$

Proof. For $m = 1, \dots, M$, we define

$$F_m \coloneqq \mathrm{clip}_R \circ f_m \circ g_m \circ \cdots \circ \mathrm{clip}_R \circ f_1 \circ g_1, \quad \widetilde{F}_m \coloneqq \mathrm{clip}_R \circ \widetilde{f}_m \circ \widetilde{g}_m \circ \cdots \circ \mathrm{clip}_R \circ \widetilde{f}_1 \circ \widetilde{g}_1,$$

and $F_0 \coloneqq \mathrm{id}, \widetilde{F}_0 \coloneqq \mathrm{id}$. Then, it holds

$$F_m = \mathrm{clip}_R \circ f_m \circ g_m \circ F_{m-1}, \quad \widetilde{F}_m = \mathrm{clip}_R \circ \widetilde{f}_m \circ \widetilde{g}_m \circ \widetilde{F}_{m-1}$$

for $m = 1, \dots, M$. Note that $\|F_m\|_\infty \le R$ and $\left\|\widetilde{F}_m\right\|_\infty \le R$ for any $m = 1, \dots, M$ due to the clipping.

For any $X \in \mathbb{R}^{d \times \infty}$ with $\|X\|_\infty \le r$ and $m = 1, \dots, M$, we have

$$\begin{aligned}
\left\|F_m(X) - \widetilde{F}_m(X)\right\|_\infty &= \left\|\mathrm{clip}_R \circ f_m \circ g_m \circ F_{m-1}(X) - \mathrm{clip}_R \circ \widetilde{f}_m \circ \widetilde{g}_m \circ \widetilde{F}_{m-1}(X)\right\|_\infty \\
&= \left\|f_m \circ g_m \circ F_{m-1}(X) - \widetilde{f}_m \circ \widetilde{g}_m \circ \widetilde{F}_{m-1}(X)\right\|_\infty \\
&\qquad (\because \mathrm{clip}_R \text{ is 1-lipschitz continuous.}) \\
&\le \left\|f_m \circ g_m \circ F_{m-1}(X) - \widetilde{f}_m \circ g_m \circ F_{m-1}(X)\right\|_\infty \\
&\quad + \left\|\widetilde{f}_m \circ g_m \circ F_{m-1}(X) - \widetilde{f}_m \circ \widetilde{g}_m \circ F_{m-1}(X)\right\|_\infty \\
&\quad + \left\|\widetilde{f}_m \circ \widetilde{g}_m \circ F_{m-1}(X) - \widetilde{f}_m \circ \widetilde{g}_m \circ \widetilde{F}_{m-1}(X)\right\|_\infty.
\end{aligned}$$

For the first term, since $\|g_m \circ F_{m-1}(X)\| \le (BDRc)^2$ due to the first argument of Lemma J.6, using the third argument of Lemma J.5, we have

$$\left\|f_m \circ g_m \circ F_{m-1}(X) - \widetilde{f}_m \circ g_m \circ F_{m-1}(X)\right\|_\infty \le 2(2BW)^L (BDRc)^2 \cdot \delta.$$

For the second term, the second argument of Lemma J.5 and the third argument of Lemma J.6 yield

$$\begin{aligned}
\left\|\widetilde{f}_m \circ g_m \circ F_{m-1}(X) - \widetilde{f}_m \circ \widetilde{g}_m \circ F_{m-1}(X)\right\|_\infty &\le (BW)^L \|g_m \circ F_{m-1}(X) - \widetilde{g}_m \circ F_{m-1}(X)\|_\infty \\
&\le (BW)^L \cdot \left(2BR^2 c + (BR)^2 \cdot \iota\right) \cdot \delta.
\end{aligned}$$

For the thrid term, the third argument of Lemma J.5 and the third argument of Lemma J.6 imply

$$\begin{aligned}
&\left\|\widetilde{f}_m \circ \widetilde{g}_m \circ F_{m-1}(X) - \widetilde{f}_m \circ \widetilde{g}_m \circ \widetilde{F}_{m-1}(X)\right\|_\infty \\
&\qquad \le (BW)^L \left\|\widetilde{g}_m \circ F_{m-1}(X) - \widetilde{g}_m \circ \widetilde{F}_{m-1}(X)\right\|_\infty \\
&\qquad \le (BW)^L \cdot \left(2B^2 Rc + BR \cdot \kappa\right) \cdot \left\|F_{m-1}(X) - \widetilde{F}_{m-1}(X)\right\|_\infty.
\end{aligned}$$

Let λ_1, λ_2 be the constants defined by

$$\begin{aligned}
\lambda_1 &\coloneqq \left(2(2BW)^L (BDRc)^2 + (BW)^L \cdot \left(2BR^2 c + (BR)^2 \cdot \iota\right)\right) \cdot \delta \\
\lambda_2 &\coloneqq (BW)^L \cdot \left(2B^2 Rc + BR \cdot \kappa\right).
\end{aligned}$$

Then, we have

$$\left\|F_m(X) - \widetilde{F}_m(X)\right\|_\infty \le \lambda_1 + \lambda_2 \cdot \left\|F_{m-1}(X) - \widetilde{F}_{m-1}(X)\right\|_\infty.$$

This implies

$$\left\|F_m(X) - \widetilde{F}_m(X)\right\|_\infty + \frac{\lambda_1}{\lambda_2 - 1} \le \lambda_2 \cdot \left(\left\|F_{m-1}(X) - \widetilde{F}_{m-1}(X)\right\|_\infty + \frac{\lambda_1}{\lambda_2 - 1}\right).$$

Thus, by induction, we have

$$\left\|F_m(X) - \widetilde{F}_m(X)\right\|_\infty + \frac{\lambda_1}{\lambda_2 - 1} \le \lambda_2^m \cdot \left(\left\|F_0(X) - \widetilde{F}_0(X)\right\|_\infty + \frac{\lambda_1}{\lambda_2 - 1}\right) = \frac{\lambda_2^m \cdot \lambda_1}{\lambda_2 - 1}.$$

Since $\lambda_2 > 1$, it holds

$$\left\|F_m(X) - \widetilde{F}_m(X)\right\|_\infty \leq \lambda_1 \cdot \frac{\lambda_2^m - 1}{\lambda_2 - 1} = \lambda_1 \cdot \left(1 + \lambda_2 + \cdot + \lambda_2^{m-1}\right) \leq m\lambda_1\lambda_2^{m-1}.$$

Now, using

$$\lambda_1 \leq 3(2BW)^L(BDRc)^2(1+\iota)\cdot\delta, \quad \lambda_2 \leq 2(2BW)^L(BDRc)^2(1+\kappa),$$

we have

$$\left\|F(X) - \widetilde{F}(X)\right\|_\infty \leq M\lambda_1\lambda_2^{M-1} \leq 2^{M+1}(2BW)^{ML}(BDRc)^{2M}(1+\kappa)^M(1+\iota)\cdot\delta,$$

which completes the proof. □

Finally, we prove Theorem J.3.

Proof of Theorem J.3.

$$\begin{aligned}
\kappa &= 0 \\
\iota &\leq 2U \cdot (2BW')^{L'} \\
c &\leq U(2BW')^{L'}
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\left\|F(X) - \widetilde{F}(X)\right\|_\infty &\leq 2^{M+1}(2BW)^{ML}(BDRU(2BW')^{L'})^{2M} \cdot \left(2 \cdot 2U(2BW')^{L'}\right) \cdot \delta \\
&= 2^{M+3}(2BW)^{ML}(2BW')^{(2M+1)L'}(BDRU)^{2M+1} \cdot \delta.
\end{aligned}$$

The number of parameters in a FNN is $2W^2L$. Additionally, the number of parameters in a gated convolution layer is $2D^2$. Moreover, the number of nonzero parameters in whole network is bounded by $M(S + 2D^2)$. Therefore, the covering number can be evaluated as

$$\begin{aligned}
&\log \mathcal{N}(\mathcal{S}(M, U, D, L, W, S, B), \delta, \|\cdot\|_\infty) \\
&\quad \leq M(S + 2D^2) \\
&\qquad + \log\left(\frac{M\left(2W^2L + D^2\right) \cdot B \cdot 2^{M+3}(2BW)^{ML}(2BW')^{(2M+1)L'}(BDRU)^{2M+1}}{\delta}\right) \\
&\quad \lesssim M^2L(S + D^2)\log\left(\frac{DULWB}{\delta}\right),
\end{aligned}$$

which completes the proof. □

J.1. Proof of Theorem 4.2

Proof of Theorem 4.2. Theorem F.1 implies that, for any $T > 0$, there exists an SSM $F \in \mathcal{S}(M, U, D, L, W, S, B)$ with

$$\begin{aligned}
&M = 1, \quad \log U \sim T, \quad D \sim T^{1/\alpha}, \quad L \sim T, \quad W \sim T^{1/\alpha}2^{T/a^\dagger}, \\
&S \sim T^{2/\alpha}\max\left\{T^{2/\alpha}, T^2\right\}2^{T/a^\dagger}, \quad \log B \sim T^{1/\alpha},
\end{aligned}$$

such that $\|F - F^\circ\|_{2,P_X} \lesssim 2^{-T}$. Therefore, it holds

$$\frac{1}{r-l+1}\sum_{t=l}^{r}\|F_i - F_i^\circ\|_{2,P_X}^2 \leq 2^{-2T}.$$

Note that, thanks to the clipping, it holds $\|F\|_\infty \leq R$, and thus this inequality gives the upper bound for the first term of the right-hand side of Proposition J.2.

Next, Theorem J.3 shows that it holds

$$\log \mathcal{N}(\mathcal{S}, \delta, \|\cdot\|_\infty) \lesssim 2^{T/a^\dagger} T^{1+2/\alpha} \max\left\{T^{3/\alpha}, T^4\right\} \log \frac{T}{\delta}.$$

Combining the above two inequalities and Proposition J.2, it holds

$$R_{l,r}(\hat{F}, F^\circ) \lesssim 2^{-2T} + \frac{2^{T/a^\dagger} T^{1+2/\alpha} \max\left\{T^{3/\alpha}, T^4\right\} \log \frac{T}{\delta}}{n} + \delta.$$

By setting $T = \frac{a^\dagger}{2a^\dagger+1} \log n$ and $\delta = 1/n$, we have

$$R_{l,r}(\hat{F}, F^\circ) \lesssim n^{-\frac{2a^\dagger}{2a^\dagger+1}} (\log n)^{2+2/\alpha} \max\left\{(\log n)^{3/\alpha}, (\log n)^4\right\}.$$

□

J.2. Proof of Theorem 4.4

Proof of Theorem 4.4. Theorem F.2 implies that, for any $T > 0$, there exists an SSM $F \in \mathcal{S}(M, U, D, L, W, S, B)$ with

$$\begin{aligned}
&M = T^{1/\alpha}, \quad U = V, \quad D \sim T^{c_{\alpha,\beta}} \log^2 V, \\
&L \sim T^{c_{\alpha,\beta}} \log^3 V, \quad W \sim 2^{T/a^\dagger} T^{c_{\alpha,\beta}} \log^2 V, \\
&S \sim 2^{T/a^\dagger} T^{c_{\alpha,\beta}} \log^3 V, \quad \log B \sim T^{c_{\alpha,\beta}} \log^3 V,
\end{aligned}$$

such that $\|F - F^\circ\|_{2,P_X} \lesssim 2^{-T}$. Therefore, it holds

$$\frac{1}{r-l+1} \sum_{t=l}^{r} \|F_i - F_i^\circ\|_{2,P_X}^2 \leq 2^{-2T}.$$

Next, Theorem J.3 shows that it holds

$$\log N(\mathcal{S}, \delta, \|\cdot\|_\infty) \lesssim 2^{T/a^\dagger} T^{2/\alpha + 4c_{\alpha,\beta}} (\log V)^{10} \log \frac{1}{\delta}.$$

As same as the proof of Theorem 4.2, we can show that

$$R_{l,r}(\hat{F}, F^\circ) \lesssim 2^{-2T} + \frac{2^{T/a^\dagger} T^{2/\alpha + 4c_{\alpha,\beta}} (\log V)^{10} \log \frac{1}{\delta}}{n} + \delta.$$

By setting $T = \frac{a^\dagger}{2a^\dagger+1} \log n$ and $\delta = 1/n$, we have

$$R_{l,r}(\hat{F}, F^\circ) \lesssim n^{-\frac{2a^\dagger}{2a^\dagger+1}} (\log n)^{1+2/\alpha+4c_{\alpha,\beta}} \log^{10} V.$$

□

K. Additional details on the experiments

All the code was implemented in Python 3.10.14 with Pytorch 1.13.1 and CUDA ver 11.7. The experiments were conducted on Ubuntu 20.04.5 with A100 PCIe 40GB.

Genomic Benchmark dataset (Grešová et al., 2023) is given with the Apache License Version 2.0 and can be accessed from `https://github.com/ML-Bioinfo-CEITEC/genomic_benchmarks`. The pretrained model of Hyena is given with the Apache License Version 2.0 and can be accessed from `https://github.com/HazyResearch/safari?tab=readme-ov-file`.

For the training and evaluation of models, we utilized the code provided at `https://colab.research.google.com/drive/1wyVEQd4R3HYLTUOXEEQmp_I8aNC_aLhL`.

Experiment 1. We used the dataset `human_enhancers_cohn` of Genomic Benchmark dataset. As for the pretrained model of Hyena, we used `hyenadna-tiny-1k-seqlen`. The model was fine-tuned for 100 epochs. Then, we sampled 20 different test sequences whose correct probability is larger or equal to 0.95. For each sequence, we repeatedly mask the tokens that maximize the correct probability. The error bar is calculated by the standard deviation of these 20 samples. The source code for the experiment is `downstream_finetune.py` and `downstream_mask.py`, which can be found in the supplemental material. Finetuning needs around one hour, and masking needs around 90 minutes.

Experiment 2. We used the dataset `demo_human_or_worm`, and we fine-tuned the model using the data labeled "human". As for the pretrained model of Hyena, we used `hyenadna-small-32k-seqlen`. The model was fine-tuned for 10 epochs. Then, we sampled 20 different test sequences that have more than 20 tokens with the correct probability > 0.35. The error bar is calculated by the standard deviation of these 20 samples. The source code for the experiment is `nextword_finetune.py` and `nextword_mask.py`, which can be found in the supplemental material. Finetuning needs around 6 hours, and masking needs around 20 minutes.