Structured State Space Models

Jack Ruder

November 19, 2025

Why Transformers are not enough

- Field of ML is strongly moving towards *foundation models*
  - A foundation model is a large pre-trained model that can be adapted to a wide range of downstream tasks
- Transformers can be adapted to many tasks, but need *specialization*
  - I.E. vision transformers, long context transformers, etc.
  - Most of these are variants of the original architecture but still utilize the same core attention mechanism
  - Really, there is a strong reliance on matching inductive biases of models to data
- Attention mechanism is fantastic for long-range modeling, but still expensive with roughly $O(T^2)$ attention matrix.

High Level Task

For input $u(t) : \mathbb{R} \to \mathbb{R}$, output $y(t)$, and state $x(t)$:

$$\begin{aligned} \frac{d}{dt}x(t) &= \mathbf{A}x(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}x(t) + \mathbf{D}u(t), \end{aligned}$$

where $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times 1}$, $\mathbf{C} \in \mathbb{R}^{1\times N}$, and $\mathbf{D} \in \mathbb{R}$.
Goal: Learn the map $u(t) \mapsto y(t)$.
Note: We aren't doing probabilistic inference here.

HiPPO

- Issue: linear first-order ODEs are solved by exponential functions.
  - Gradients scale as $e^{\mathbf{A}t}$, which can explode or vanish quickly.
  - This leads to memory-loss.
- Given $u(t) \in \mathbb{R}$ for $t \geq 0$, we wish to approximate the cumulative history

$$u_{\leq t} := u(\tau)|_{\tau \leq t}$$

- *Some* compression is required, since the space of functions is uncountable

Function Approximation

Recall that we can compare two functions according to their L^2 inner product w.r.t probability measure μ:

$$\langle f, g \rangle = \int_0^\infty f(x)g(x)d\mu(x).$$

- Natural approximations in this space are given by projections onto orthogonal bases, i.e. take the truncated Fourier series with N terms.
- The probability measure of interest will weight inputs by their importance.

- Let $u(t)$ be an input function, and $\omega(t)$ a fixed probability measure. Assume we have a basis of N functions $\{p_n\}_{n=0}^{N-1}$ that are orthogonal with respect to $\omega(t)$.
- We think of $\omega(t)$ as a weighting function that emphasizes recent inputs more than older ones (e.g., exponential decay).
- The projection of $u(t)$ onto this basis with respect to $\omega(t)$ is given by

$$x_n(t) = \int u(\tau) p_n(\tau) \omega(\tau) d\tau.$$

- This gives an optimal N-term approximation of $u(t)$ in the weighted L^2 space defined by $\omega(t)$.
- **The dynamics of $x_n(t)$ can be expressed as a linear ODE**

The scaled Legendre measure (LegS) assigns uniform weighting to all history, $\mu^{(t)} = \frac{1}{t}\mathbb{I}_{[0,t]}$

Theorem

For time-invariant linear ODE

$$\frac{d}{dt}c(t) = -\frac{1}{t}\mathbf{A}c(t) + \frac{1}{t}\mathbf{B}f(t),$$

$$A_{nk} = \begin{cases} (2n+1)^{1/2}(2k+1)^{1/2}, & n > k \\ n+1, & n = k \\ 0, & n < k \end{cases}$$

$$B_n = (2n+1)^{1/2}$$

LSSL: Linear State Space Layer

- For SSM with $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ as before, impose representation of $x(t)$ as the coefficients of the projection, and choose $\mathbf{A}, \mathbf{B}$ accordingly.
- By specifying step size Δt, the SSM is discretized into a linear recurrence, and is thus simulated as a stateful recurrent model. That is, this can generalize RNNs.
- In practice, for use in a deep network, we stack K layers, and broadcast $\mathbf{B}, \mathbf{C}, \mathbf{D}, \Delta t$ appropriately.
- Can't really learn $\mathbf{A}, \Delta t$ (but these are crucial to get right).
  - On sequential MNIST, random $\mathbf{A}$ compared to HiPPO $\mathbf{A}$ gives 60% vs 98% accuracy.

Structured State Space Models

- Discretization of LSSLs are expensive to compute (due to recurrence), but this is vectorized by rewriting the recurrence as a discrete convolution.
  - $O(N \log N)$ via FFTs
  - Only efficient if we know the convolution kernel $K \in \mathbb{R}^L$ in closed form.
- Solution: diagonalize
- Result: Conjugation is an equiv. relation on SSMs: $(\mathbf{A}, \mathbf{B}, \mathbf{C}) \sim (\mathbf{V}^{-1}\mathbf{A}\mathbf{V}, \mathbf{V}^{-1}\mathbf{B}, \mathbf{C}\mathbf{V})$
- That is, both SSMs represent the same $u(t) \mapsto y(t)$ under a change of basis.

Diagonalization sucks

- The HiPPO matrix is diagonalizable, but entries of the matrix scale exponentially in N.
- Instead, A ideally should be conjugated by nice unitary matrices, so A ideally is normal (orthogonal eigenbasis).
- However, HiPPO A is not normal.
- Solution: Approximate HiPPO with a normal matrix plus a low-rank matrix.
- Proceed to find K by linear algebra, see paper.

How good is S4?

- Comparable to transformers, but $O(N + L)$ in memory.
- Often can outperform transformers, except for NLP tasks (for which transformers were designed).
- Suffers from numerical instability due to poor parameter initialization
- Can be designed to be extremely well-optimized in hardware, but this requires appropriate hardware for full benefits.

Successors to S4

- S5: Selective S4, adds hierachical gating mechanisms to limit information the model must see.
- Mamba: Fixes S4 to be specifically hardware-aware, and handles the initialization issues, also uses gating.
- Jamba: Mixture of experts, i.e Mamba + Transformer
- Mamba 2: Massively improves upon Mamba **and shows that these *almost* generalize linear attention**

In fact, Mamba 2 improves upon linear attention by avoiding memory-collision issues, and combining ideas from both sets of literature (SSSM/Transformers)