Paper

https://arxiv.org/ftp/arxiv/papers/2312/2312.00752.pdf

Intro

- Transformer and attention layer has cons
    - an inability to model anything outside of a finite window
    - Quadratic scaling
    - Fixing these issues comes at an expense and not completely scalable
- Structured State Space Models
    - A combination of cnn and rnn, and inspiration from kalman state space models

Paper Improvements

- Paper introduces a new class of selective state space models that achieves the modeling power of Transformers while scaling linearly in sequence length.
- Selection Mechanism - allow model to filter out irrelevant info
- Hardware-aware Algorithm - computes the model recurrently with a scan instead of convolution
- Architecture
    - Combines prior SSM architectures with mlp block of transformer into a single block
- Mamba is the first linear-time sequence model that achieves Transformer-quality performance in downstream and pretraining evaluations
- Scales up to 1B parameters

Structured State Space Sequence Models (S4)

- a recent class of sequence models for deep learning that are broadly related to RNNs, and CNNs, and classical state space models
- Have 4 parameters Δ, A, B, C
- Δ, A, B, C to $\bar{A}\ \bar{B}\ C$
    - Transform ‘continuous’ parameters to ‘discrete’
- After above step, model can be computed in a linear recurrence or a global convolution
- Has 2 stages: discretization and computation

$$h'(t) = \boldsymbol{A}h(t) + \boldsymbol{B}x(t) \quad (1a) \qquad h_t = \overline{\boldsymbol{A}}h_{t-1} + \overline{\boldsymbol{B}}x_t \quad (2a) \qquad \overline{\boldsymbol{K}} = (C\overline{\boldsymbol{B}}, C\overline{\boldsymbol{AB}}, \dots, C\overline{\boldsymbol{A}}^{k}\overline{\boldsymbol{B}}, \dots) \quad (3a)$$

$$y(t) = \boldsymbol{C}h(t) \quad (1b) \qquad y_t = \boldsymbol{C}h_t \quad (2b) \qquad y = x * \overline{\boldsymbol{K}} \quad (3b)$$

$$\overline{\boldsymbol{A}} = \exp(\Delta \boldsymbol{A}) \qquad \overline{\boldsymbol{B}} = (\Delta \boldsymbol{A})^{-1}(\exp(\Delta \boldsymbol{A}) - \boldsymbol{I}) \cdot \Delta \boldsymbol{B}$$

Other SSM Architectures

- Rmkv
- H3
- Hyena
- RetNet
-

SSM and Selection

- We argue that a fundamental problem of sequence modeling is compressing context into a smaller state.
- For example, attention is both effective and inefficient because it explicitly does not compress context at all. This can be seen from the fact that autoregressive inference requires explicitly storing the entire context (i.e. the KV cache), which directly causes the slow linear-time inference and quadratic-time training of Transformers
- The Selective Copying task modifies the popular Copying task (Arjovsky, Shah, and Bengio 2016) by varying the position of the tokens to memorize. It requires content-aware reasoning to be able to memorize the relevant tokens (colored) and filter out the irrelevant ones (white).

- The Induction Heads task is a well-known mechanism hypothesized to explain the majority of in-context learning abilities of LLMs (Olsson et al. 2022). It requires context-aware reasoning to know when to produce the correct output in the appropriate context (black).

Copy Problem

- Recurrent networks have been known to have trouble remembering information about inputs seen many time steps previously

Improving SSM with selection

- One method of incorporating a selection mechanism into models is by letting their parameters that affect interactions along the sequence (e.g. the recurrent dynamics of an RNN or the convolution kernel of a CNN) be input-dependent.

Efficient SSM with selection

Linear Time invariance