Structured State Space (S4) Model: Implementation and Evaluation on sCIFAR-10 Dataset

Aarush Rathore

April 5, 2025

1 Introduction

Structured State Space (S4) models are a recent advancement in sequence modeling, addressing the computational inefficiencies of traditional architectures like Transformers. S4 leverages state-space models (SSMs) with efficient parameterizations and discretization techniques to handle long-range dependencies in sequences. This document provides an overview of the implementation, data preparation, and evaluation of the S4 model on the sequential CIFAR-10 (sCIFAR-10) dataset.

2 Data Preparation for sCIFAR-10

The CIFAR-10 dataset consists of 60,000 images, each of size $32 \times 32 \times 3$, divided into 50,000 training images and 10,000 test images across 10 classes. To adapt CIFAR-10 for sequence modeling:

- Each image was flattened into a sequence of length 1024 with 3 channels.
- Transformations included:
  - Conversion to tensors.
  - Reshaping to $(1024, 3)$ using a custom function.
- Data subsets were optionally created by sampling a fraction (20%, 50%, or 100%) of the training data for faster experimentation.
- PyTorch DataLoaders were used for efficient batching and shuffling during training and evaluation.

This preprocessing ensured compatibility with the S4 model's sequence-based architecture.

3 Key Components of the S4 Model

The S4 model builds upon several innovations in state-space modeling:

- **HiPPO Framework**: The HiPPO matrix is a lower triangular matrix that compresses input history efficiently. It initializes the state matrix A in SSMs to approximate long-range dependencies using Legendre polynomials.
- **DPLR Parameterization**: The state matrix A is decomposed into a diagonal-plus-low-rank (DPLR) form to enable efficient computation of the convolution kernel.
- **Discretization Techniques**: Three methods were implemented:
  - Zero-Order Hold (ZOH): A simple discretization method.
  - Bilinear Transform: Uses Tustin's method for stability.
  - Generalized Bilinear Transform: Introduces a tunable parameter ($\alpha = 0.5$).
- **S4Block**: Combines SSMs with normalization, activation (GELU), and residual connections.
- **S4Model**: Stacks multiple S4Blocks with an encoder and decoder for sequence-to-sequence transformations.

These components allow S4 to efficiently model long sequences while maintaining computational efficiency.

4 Experiment Details and Results

Four configurations were tested on sCIFAR-10 using different data subsets, model architectures, and discretization schemes. Below are the details:

4.1 SSM1.ipynb

- **Conditions**: - Dataset: 50% of CIFAR-10 training data. - Model Architecture: 2 layers, $d_{\text{model}} = 64$. - Discretization Schemes: ZOH, Bilinear Transform, Generalized Bilinear Transform. - Training: 5 epochs. - **Results**: - ZOH: Best test accuracy = **34.71**%. - Bilinear: Best test accuracy = **24.69**%. - Generalized: Best test accuracy = **32.43**%.

4.2 SSM2.ipynb

- **Conditions**: - Dataset: 20% of CIFAR-10 training data. - Model Architecture: 3 layers, $d_{\text{model}} = 64$. - Discretization Schemes: ZOH, Bilinear Transform, Generalized Bilinear Transform. - Training: 5 epochs. - **Results**: - ZOH: Best test accuracy = **28.50**%. - Bilinear: Best test accuracy = **25.07**%. - Generalized: Best test accuracy = **26.23**%.

4.3 SSM3.ipynb

- **Conditions**: - Dataset: Full CIFAR-10 training data (100%). - Model Architecture: 3 layers, $d_{\text{model}} = 64$. - Discretization Scheme: Bilinear Transform only. - Training: 5 epochs. - **Results**: - Bilinear: Best test accuracy = **30.39**%.

4.4 SSM4.ipynb

- **Conditions**: - Dataset: Full CIFAR-10 training data (100%). - Model Architecture: 4 layers, $d_{\text{model}} = 128$. - Discretization Scheme: Bilinear Transform only. - Training: Up to 50 epochs with early stopping enabled. - **Results**: - Bilinear: Best test accuracy = **28.74**%. Early stopping was triggered at epoch 10.

Summary Table

The table below summarizes results across all configurations:

Notebook	Data Usage	Layers	Discretization Schemes	Best Accuracy (%)
SSM1.ipynb	50%	2	ZOH, Bilinear, Generalized	ZOH: **34.71**
SSM2.ipynb	20%	3	ZOH, Bilinear, Generalized	ZOH: **28.50**
SSM3.ipynb	Full (100%)	3	Bilinear	Bilinear: **30.39**
SSM4.ipynb	Full (100%)	4	Bilinear	Bilinear: **28.74**

Table 1: Summary of experiment results across different configurations.

From these results, it is evident that the choice of discretization scheme and dataset size significantly impacts performance.

5 Conclusion

The S4 model demonstrates strong potential as an efficient sequence modeler for long-range tasks(though I was not able to replicate them exactly but various notebooks trained show **Proof Of Concept** for the given project). Key takeaways include: - The HiPPO framework plays a critical role in modeling long-range dependencies effectively. - Among discretization schemes tested, ZOH achieved the highest accuracy (34.71%) in this study. - Increasing dataset size generally improved performance but required careful tuning of hyperparameters.

Future work could explore additional datasets or further optimize discretization methods to enhance performance.I tried to experiment with various permutations to see how these models perform on changing parameters but was not able to fully get any coherent view.Being GPU Poor I was restricted by compute time.In future I intend to train them with more epochs and layers .