# Sparse Modular Activation for Efficient Sequence Modeling

***Liliang Ren***[1,2], *Yang Liu*[2], *Shuohang Wang*[2], *Yichong Xu*[2],
*Chenguang Zhu*[2], *ChengXiang Zhai*[1]
University of Illinois Urbana Champaign[1]; Microsoft[2]
liliang3@illinois.edu

Great Hall & Hall B1+B2 #508 @NeurIPS 2023

# Motivation

Attention-free Sequence Model:

- Linear State Space Models (SSMs) [Gu et al., 2022]

Previous Works:

- Statically combine attention with SSMs [Ma et al., 2023; Dao et al., 2023]
    - Over-assuming attention modules are needed for all sequence elements.

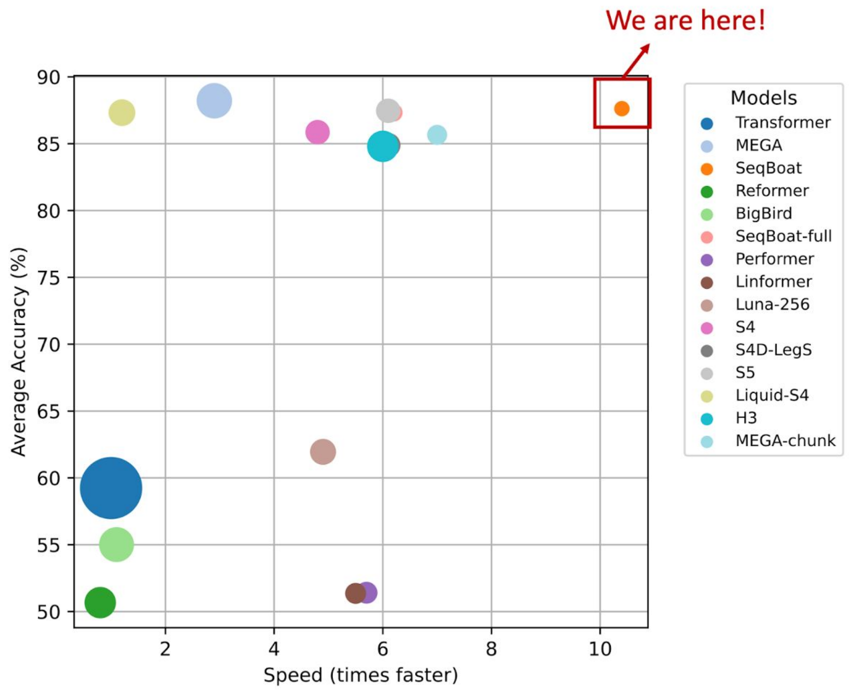
Research Questions:

- **RQ1:** Can neural networks learn to activate their attention modules **on-demand** for better efficiency?
- **RQ2:** How much extra attention is needed on a task-by-task basis?

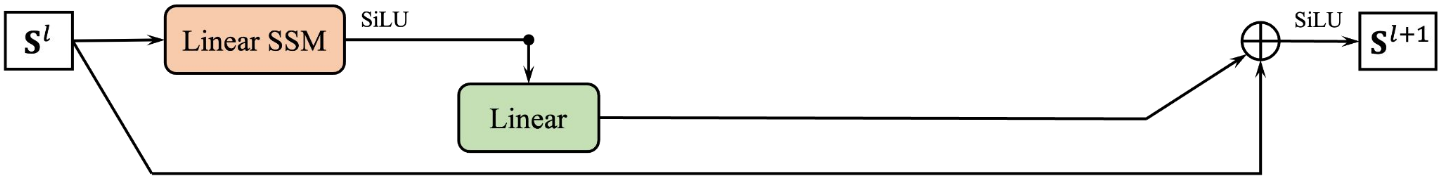
# Contribution

- Sparse Modular Activation (SMA)
  - General activation mechanism for modules
  - **Theoretical guarantee** of space coverage
  - **Efficient** parallel implementation

- SeqBoat: A novel architecture with SMA
  - SSMs + Sparsely Activated GAU [Hua et al., 2022]
  - SoTA quality-efficiency trade-off on LRA → **RQ1**
  - Reveals attention needed for each data sample and task → **RQ2**

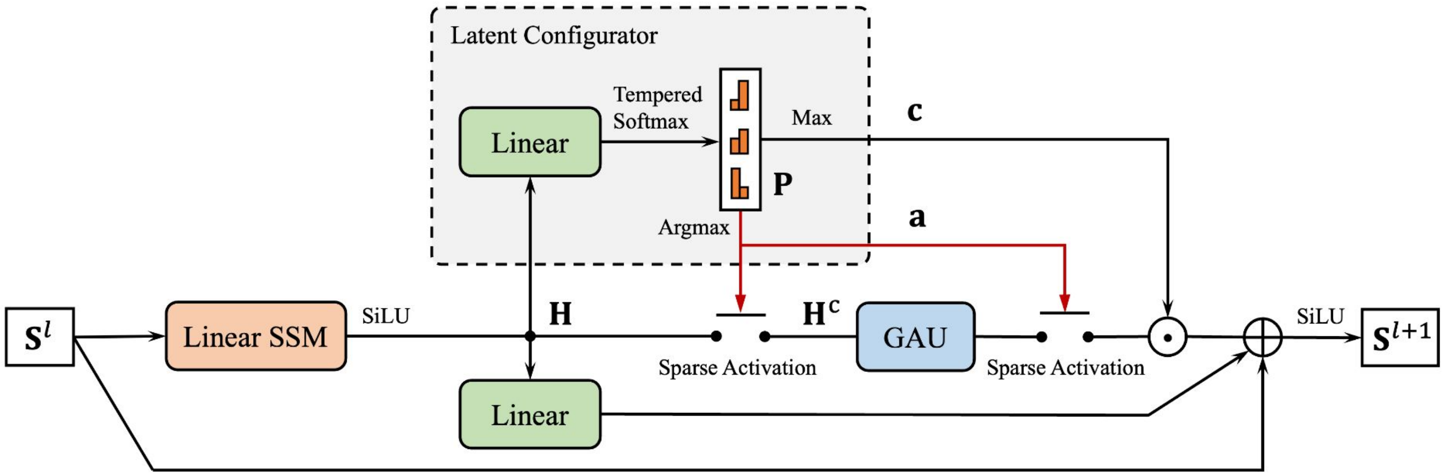


# Base Architecture



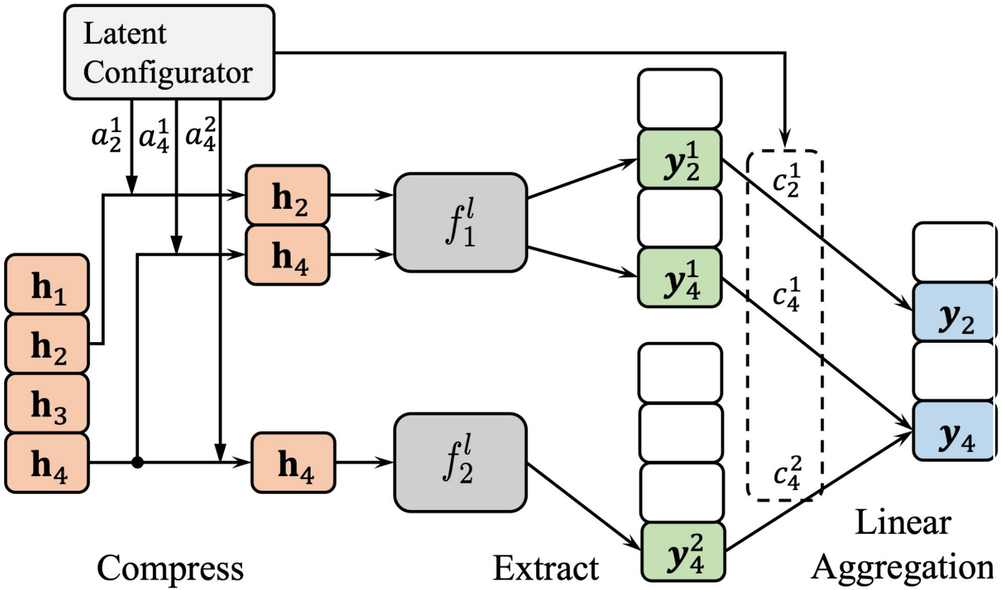

- An SSM layer with SiLU non-linearity and skip connection
- We use MD-EMA [Ma et al., 2023] for SSM's kernel parametrization

# SeqBoat Architecture 🚤




- Latent Configurator decides if a GAU is needed for each time step independently.
- No Feed Forward Network (FFN)

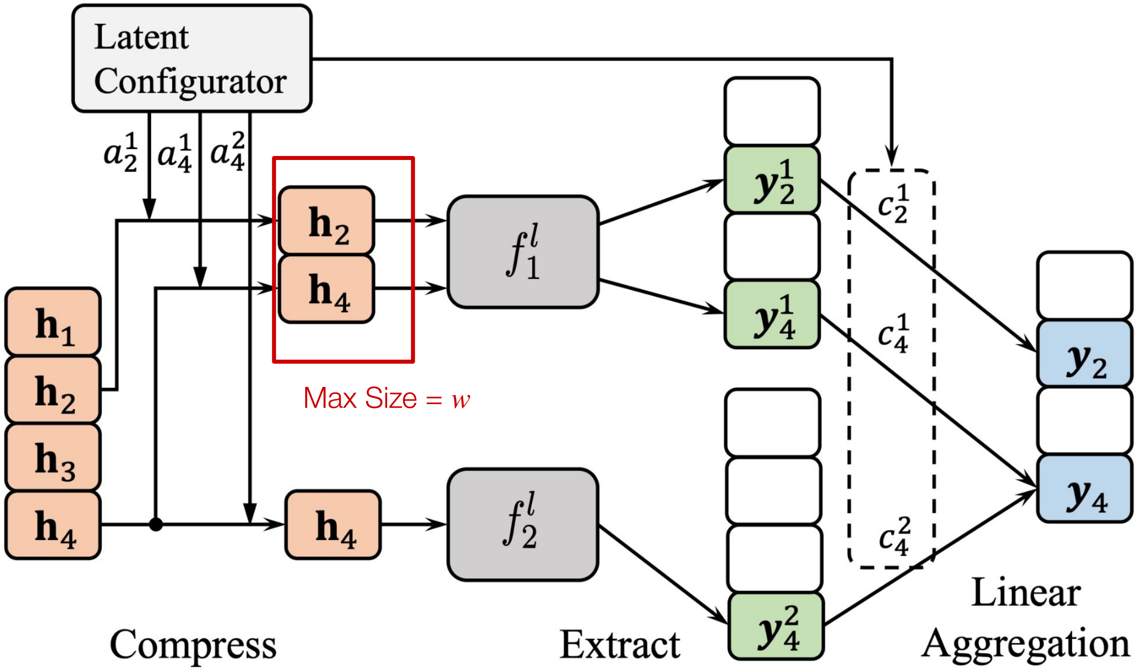
# Sparse Modular Activation (SMA)




- Given actions ***a*** and confidences ***c***, SMA **efficiently** compresses inputs and extracts outputs in parallel.
- SMA is proved to have a **full coverage** of the function space over modules.

**Theorem 1** (Function Space Coverage of SMA). *For any $\mathcal{L}' \subseteq \mathcal{L} = span\{f_1^l, ..., f_M^l\}$, there exists a pair of $(\mathbf{a}_t', \mathbf{c}_t')$ that $\mathcal{L}_{SMA}(\mathbf{a}_t', \mathbf{c}_t') = \mathcal{L}'$. In other words, SMA has a* ***full*** *coverage of the function space $\mathcal{L}$.*

# Working Memory Mechanism




- Keeps a First-In-First-Out history for each module with size $w$.
- For GAU, it is equivalent to have a local attention with window size $w$.
- Reduces worst-case time complexity to **linear**.
- Still allows interaction between far-apart inputs.

# Long Range Arena

| Models | ListOps | Text | Retr. | Image | Path. | Path-X | Avg. | Speed | Mem. |
|---|---|---|---|---|---|---|---|---|---|
| *Quadratic Inference Complexity* | | | | | | | | | |
| Transformer | 37.11 | 65.21 | 79.14 | 42.94 | 71.83 | ✗ | 59.24 | 1× | 1× |
| MEGA* | 63.14 | 90.43 | 91.25 | 90.44 | 96.01 | 97.98 | 88.21 | 2.9× | 0.31× |
| *Sub-quadratic Inference Complexity* | | | | | | | | | |
| Reformer | 37.27 | 56.10 | 53.40 | 38.07 | 68.50 | ✗ | 50.67 | 0.8× | 0.24× |
| BigBird | 36.05 | 64.02 | 59.29 | 40.83 | 74.87 | ✗ | 55.01 | 1.1× | 0.30× |
| **SeqBoat-full*** | 61.65 | 89.60 | 91.67 | 89.96 | 95.87 | 95.28 | 87.33 | 6.2× | 0.07× |
| *Linear Inference Complexity* | | | | | | | | | |
| Performer | 18.01 | 65.40 | 53.82 | 42.77 | 77.05 | ✗ | 51.41 | 5.7× | 0.11× |
| Linformer | 35.70 | 53.94 | 52.27 | 38.56 | 76.34 | ✗ | 51.36 | 5.5× | 0.10× |
| Luna-256 | 37.98 | 65.78 | 79.56 | 47.86 | 78.55 | ✗ | 61.95 | 4.9× | 0.16× |
| S4 | 59.10 | 86.53 | 90.94 | 88.48 | 94.01 | 96.07 | 85.86 | 4.8× | 0.14× |
| S4D-LegS | 60.47 | 86.18 | 89.46 | 88.19 | 93.06 | 91.95 | 84.89 | 6.1× | 0.14× |
| S5 | 62.15 | 89.31 | **91.40** | 88.00 | 95.33 | **98.58** | 87.46 | 6.1× | 0.14× |
| Liquid-S4 | **62.75** | 89.02 | 91.20 | 89.50 | 94.8 | 96.66 | 87.32 | 1.2× | 0.17× |
| H3* | 57.50 | 88.20 | 91.00 | 87.30 | 93.00 | 91.80 | 84.80 | 6.0× | 0.24× |
| MEGA-chunk* | 58.76 | **90.19** | 90.97 | 85.80 | 94.41 | 93.81 | 85.66 | 7.0× | 0.09× |
| **SeqBoat*** | 61.70 | 89.60 | 91.28 | **90.10** | **96.35** | 96.68 | **87.62** | 10.4× | 0.05× |

- Training Speed/Memory Allocation on Text with 4k input length
- Substantially outperforms previous hybrid models.
- State-of-The-Art Accuracy-efficiency Trade-off

# Speech Recognition and Language Modeling

- Speech Commands 10 (16k input length)

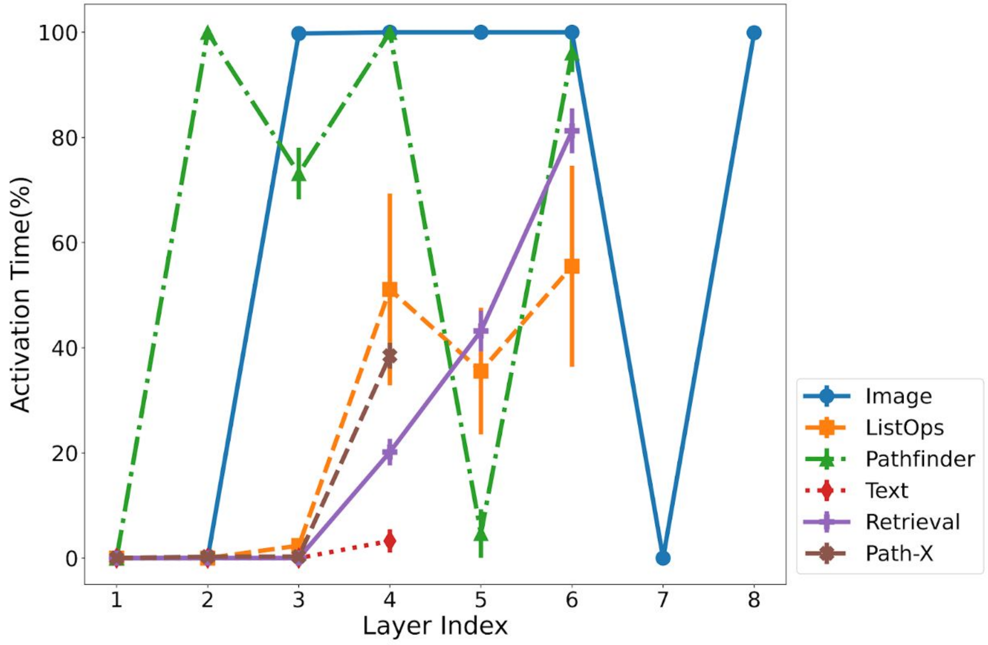
| Model | #Param. | Acc. (↑) | Speed (↑) | Mem. (↓) |
|---|---|---|---|---|
| S4 | 300K | **97.50** | - | - |
| MEGA-chunk | 300K | 96.92 | 1.00× | 1.00× |
| **SeqBoat** | 293K | 97.35 | 1.32× | 0.44× |

- enwik8 (8k input length)

| Model | #Param. | bpc (↓) | Speed (↑) | Mem. (↓) |
|---|---|---|---|---|
| Transformer-XL | 41M | 1.06 | - | - |
| Adaptive Span | 39M | 1.02 | - | - |
| MEGA-chunk | 39M | 1.02 | 1.00× | 1.00× |
| **SeqBoat** | 39M | 1.02 | 1.16× | 1.07× |

- SeqBoat offers significantly better speed-quality trade-off than MEGA-chunk

# Amount of Attention Needed per Task




- Image-based tasks need more attention than text-based tasks
- Higher layers need more attention than lower layers
- More variance of difficulty per data sample in ListOps than other tasks

# Summary

More in our paper:

- More analyses, Theorems & Proofs
- Ablation studies & Implementation details

Code Link

Conclusion:

- SMA - First mechanism enables efficient activation of a self-attention module
  - Plus theoretical guarantee of module space coverage
- SeqBoat - SoTA quality-efficiency trade-off on LRA with intrinsic interpretability

# Thanks for your time!

## Sparse Modular Activation for Efficient Sequence Modeling

*Liliang Ren, Yang Liu, Shuohang Wang, Yichong Xu, Chenguang Zhu, ChengXiang Zhai*

Great Hall & Hall B1+B2 #508 @NeurIPS 2023

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Microsoft