Never Train from Scratch: Fair Comparison of Long-Sequence Models Requires Data-Driven Priors (Extended Abstract)*

Ido Amos[1] , **Jonathan Berant**[1] , **Ankit Gupta**[2]
[1]Tel Aviv University
[2]IBM Research
idoamos@mail.tau.ac.il, joberant@cs.tau.ac.il, ankitgupta.iitkanpur@gmail.com

Abstract

This paper is an extended abstract of our ICLR 2024 Outstanding Paper Award work. Modeling long-range dependencies across sequences is a longstanding goal in machine learning. While state space models reportedly outperform Transformers on benchmarks like Long Range Arena, we show that random initialization significantly overestimates architectural differences. Pretraining with standard denoising objectives on downstream task data leads to dramatic gains across architectures and minimal performance gaps between Transformers and state space models (SSMs). We demonstrate that properly pretrained vanilla Transformers match S4 performance on Long Range Arena and improve SSM results on PathX-256 by 20 absolute points. Our analysis shows previously-proposed structured parameterizations for SSMs become largely redundant with pretraining. When evaluating architectures on supervised tasks, incorporating data-driven priors via pretraining is essential for reliable performance estimation.

1 Introduction

Self-supervised pretraining is widespread across machine learning, with pretrained model finetuning now standard practice for downstream tasks [Touvron *et al.*, 2023; Baevski *et al.*, 2020; Reed *et al.*, 2022; Raffel *et al.*, 2020]. However, when developing architectures with better inductive biases for specific skills, training from scratch with random initialization remains common [Tay *et al.*, 2021; Delétang *et al.*, 2023; Velickovic *et al.*, 2022; Dwivedi *et al.*, 2022]. This practice stems from computational constraints and attempts to enable fair comparisons without requiring a standard pretraining corpus.

Long Range Arena (LRA) exemplifies this pattern, with Transformers showing poor performance on these stress tests [Tay *et al.*, 2021], spurring development of new architectures biased toward capturing long-range dependencies [Gu *et al.*, 2022; Gupta *et al.*, 2022a; Li *et al.*, 2023; Ma *et al.*, 2023]. These results contradict the success of pretrained Transformers on long-range tasks like text summarization and protein folding [Touvron *et al.*, 2023; Jumper *et al.*, 2021]. Despite advances in long sequence modeling, the reasons for this discrepancy remain unexplored, while competitive methods rely on architectural modifications [Ma *et al.*, 2023; Zuo *et al.*, 2022].

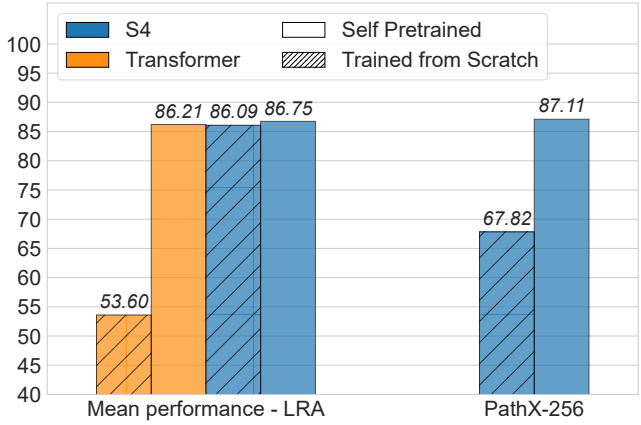

Figure 1: Evaluation of Transformers and S4 on Long Range Arena when trained from scratch vs. when self pretrained.

We demonstrate this discrepancy stems from inadequate training and evaluation practices and propose a simple solution. While avoiding large-corpus pretraining is understandable, random initialization overlooks how pretraining objectives create beneficial inductive biases. Recent work shows pretraining solely on downstream training data (self pretraining or SPT) often yields gains comparable to large-corpus pretraining [El-Nouby *et al.*, 2021; He *et al.*, 2022; Krishna *et al.*, 2023]. This suggests SPT offers a more realistic performance estimate while serving as a data-driven initialization method that enables fair comparisons using only the task data.

Our empirical evidence shows that priors learned through SPT with denoising objectives effectively capture long-range dependencies across architectures, often eliminating the need for complex hand-crafted modeling biases [Gu *et al.*, 2022;

*Ido Amos, Jonathan Berant, and Ankit Gupta. Never train from scratch: Fair comparison of long-sequence models requires data-driven priors. In *The Twelfth International Conference on Learning Representations*, 2024.

Approach	Listops	Text	Retrieval	Image	Pathfinder	PathX	Avg.
Transformer	36.37	64.27	57.46	42.44	71.40	✗	53.66
Local Attention	15.82	52.98	53.39	41.46	66.63	✗	46.71
Longformer	35.63	62.85	56.89	42.22	69.71	✗	52.88
Linformer	35.70	53.94	52.27	38.56	76.34	✗	51.14
Reformer	37.27	56.10	53.40	38.07	68.50	✗	50.56
BigBird	36.05	64.02	59.29	40.83	74.87	✗	54.17
Linear Trans.	16.13	65.90	53.09	42.34	75.30	✗	50.46
Performer	18.01	65.40	53.82	42.77	77.05	✗	51.18
Transformers + Masked SPT	**59.75**	**89.27**	88.64	74.22	88.45	87.73	81.34
Transformers + Causal SPT	59.15	88.81	**90.38**	**76.00**	**88.49**	**88.05**	**81.81**

Table 1: **Long Range Arena**. (top) performance of models trained from scratch as reported in [Tay *et al.*, 2021], (bottom) performance of self pretrained (SPT) Transformers of sizes *comparable* to the ones on top. ✗ denotes chance accuracy.

Ma *et al.*, 2023; Li *et al.*, 2023; Orvieto *et al.*, 2023]. On Long Range Arena (LRA), SPT improves vanilla Transformer performance by more than 30%, allowing them to match state-of-the-art results without architectural changes (Figure 1), contradicting prior works showing significantly lower Transformer performance.

SPT also benefits State Space models (SSMs), which use modified linear RNNs in place of attention. S4 achieves impressive performance on long sequence tasks through specialized parameterization and initialization [Gu *et al.*, 2022]. With SPT, S4 shows gains in 5 of 6 LRA tasks and solves the challenging PathX-256 task with 20% improved accuracy (Figure 1). Our analysis reveals that data-driven priors from SPT make many hand-crafted modeling biases redundant [Gupta *et al.*, 2022b], enabling competitive performance with simple diagonal linear RNNs without manual modifications [Orvieto *et al.*, 2023].

Our findings demonstrate that beneficial priors for capturing distant dependencies can be learned directly from task data through standard denoising objectives. The benefits of SPT become more pronounced with scarcer data. For SSMs, our analysis of convolution kernels reveals that, depending on the modality, rapidly decaying kernels sometimes outperform the slowly decaying ones used in native S4, highlighting the advantages of learning data-specific priors [Gu *et al.*, 2020].

Our main contributions are:

(i) Demonstrating that reported performance on long-range benchmarks is severely underestimated, and proposing an inexpensive data-driven approach for accurate evaluation, without requiring any additional data.

(ii) Reporting significant empirical gains across architectures on LRA, improving the best reported accuracy on PathX-256 by 20 absolute points ($67 \rightarrow 87$).

(iii) Showing that manually-designed biases become redundant with pretraining, enabling simpler models to match sophisticated architectures. We achieve competitive performance on LRA with Transformers and diagonal linear RNNs.

The substantial improvements from SPT across LRA's multi-modal tasks suggest including pretraining when evaluating models for multidimensional inputs [Nguyen *et al.*, 2022], algorithmic reasoning [Diao and Loynd, 2023], or graphs [Shirzad *et al.*, 2023].

Our code & data are available at https://github.com/IdoAmos/not-from-scratch.

2 Experimental Setup

We evaluate Transformers and SSMs on Long Range Arena (LRA), a benchmark for testing long-range dependency modeling [Tay *et al.*, 2021]. It contains 6 sequence classification tasks:

1. ListOps: Nested lists with operations (MAX, MEAN, etc.) applied to multiple token arguments [Nangia and Bowman, 2018]. 10-way classification with 2K sequence length. **INPUT:** `[MAX 4 3[MIN 2 3]1 0[MEDIAN 1 5 8 9 2]]` **OUTPUT:** `5`
2. Text: Character-level IMDb reviews [Maas *et al.*, 2011]. Binary sentiment classification with up to 2048 sequence length.
3. Retrieval: Character-level AAN dataset [Radev *et al.*, 2009] for document similarity. Binary classification with 8K total tokens.
4. Image: Flattened grayscale CIFAR10 images as 1D sequences. 10-way classification with 1024 sequence length.
5. Pathfinder, PathX: Synthetic visual path-tracing tasks [Linsley *et al.*, 2018; Kim *et al.*, 2020]. Binary classification with lengths 1024 and 16384.

We also test on PathX-256 (sequence length $256^2 = 65536$) and additional datasets described in Section 3.4.

Self Pretraining (SPT) We perform SPT using only the downstream task training set, with causal/autoregressive objectives for unidirectional models and masked modeling for bidirectional models. Masking ratios: 50% for visual tasks [He *et al.*, 2022], 15% for language tasks [Liu *et al.*, 2019], and 10% for ListOps. We use FLASH attention [Dao *et al.*, 2022] for Transformers, with blocked attention (block size 4096) for sequences over 16K. Our code builds on the official S4 repository. For details on hyperparameters and compute, refer to the original paper [Amos *et al.*, 2024].

3 Results

Section 3.1 shows SPT results on LRA with official configurations. Section 3.2 compares SPT for Transformers and S4. Section 3.3 evaluates SSM design choices with SPT. Section 3.4 presents additional experiments on different modalities.

Approach	Listops	Text	Retrieval	Image	Pathfinder	PathX	PathX-256	Avg.
Transformers + Rotary	47.90	79.08	82.31	75.04	76.64	84.72	✗	74.28
Transformers + Rotary + Masked SPT	61.49	**91.02**	**91.57**	86.04	94.16	92.98	✗	86.21
S4 [Gu *et al.*, 2022]	59.60	86.82	90.90	88.65	94.20	96.35	67.82†	86.09
S4 + Masked SPT	61.25	90.34	88.74	89.36	94.92	96.94	**87.11**	86.75
SPADE [Zuo *et al.*, 2022]	60.50	90.69	91.17	88.22	**96.23**	97.60	❑	87.40
MEGA [Ma *et al.*, 2023]	**63.14**	90.43	91.25	**90.44**	96.01	**97.98**	❑	**88.21**
Pythia 70M (Rand Init)	41.20	69.29	76.45	52.55	74.31	✗	✗	62.76
Pythia 70M	43.05	83.41	84.29	67.41	80.05	✗	✗	68.04

Table 2: **Long Range Arena.** Self pretrained (SPT) Transformers and S4 compared to existing trained from scratch models. Average performance ("Avg.") is reported without PathX-256 to align with prior work. Results for MEGA, SPADE & S4 are taken from original papers with exceptions denoted by †. ✗ denotes computationally infeasible, ❑ denotes unreported results.

3.1 Underestimation of Long-range Abilities of Transformers

We investigate the reliability of historically-reported LRA model performances with pretraining. We repeat the Transformer experiments from [Tay *et al.*, 2021], first pretraining models on task data before finetuning, strictly following their configurations. We test both next token prediction and masked token prediction objectives with masking ratios as described in Section 2.

As Table 1 shows, both pretraining objectives dramatically improve Transformer performance compared to random initialization, with average test accuracy increasing by roughly 30%. Causal and masked pretraining yield similar results even for visual tasks and ListOps (where arguments are randomly sampled, making token prediction from context difficult). Since we made no architectural changes and used no additional data, these improvements come entirely from priors learned during SPT, demonstrating its importance for reliable evaluation.

3.2 Comparing S4 and Transformers

In the above set-up we strictly adhered to the model sizes used by [Tay *et al.*, 2021] and consequently the absolute performances are still low compared to the current state-of-the-art on LRA. In this section, we scale the model sizes and evaluate the utility of SPT for the best performing architectures including S4 [Gu *et al.*, 2022]. For Transformers, we replace the positional embeddings with the more commonly used rotary embeddings [Su *et al.*, 2021] and only train bidirectional models in line with prior works reporting high performance.

As summarized in Table 2, SPT leads to dramatic performance gains for Transformers with performance gains ranging from $8-15\%$ across tasks, even surpassing the average performance of a well-tuned S4 (86.2 vs 86.1). SPT Transformers surpass the performance of both trained from scratch and SPT versions of S4 on 3 out of 6 tasks. The results in Table 2 defy current understanding, with prior works citing the subpar LRA performance of Transformers as a prime motivating factor for new methods. Yet we show that, while architectural developments indeed lead to remarkable performance gains, most of the priors essential to high performance can already be learned from data directly.

In case of S4, while SPT leads to modest gains on most tasks, a substantial gain of 20% is observed on the challenging PathX-256 task with input length of $65K$, significantly improving over the best reported performance of 63.1% by [Dao *et al.*, 2022] who, in addition, used extra data from the Pathfinder-64 task.

The additionally reported models, SPADE and MEGA, are Transformer variants that augment the model with a single or several state space layers. SPADE combines the outputs of a frozen S4 layer and local attention in the first block, while MEGA incorporates a learned exponential moving average, an instance of diagonal SSMs, into gated attention blocks. To the best of our knowledge, we are the first to show that purely attention-based methods, without any architectural modifications, can achieve competitive results on LRA. While incorporating SSMs can be important in terms of scalability to longer sequences due to their log-linear complexity with respect to input length, we show that in terms of model performance, pretraining leads to biases that are as effective as manual designs.

An important aspect of SPT is the use of additional compute compared to the trained from scratch baseline and it is natural to investigate if similar gains can be obtained by training from scratch for longer. For all our trained from scratch baselines, we ensured that the validation performance had converged and did not improve for several consecutive epochs. We examine the aspect of the computational overhead of SPT in detail in the original paper [Amos *et al.*, 2024], where we show that SPT leads to significant gains, even in the setting where the same amount of compute is used for SPT models and the ones that are trained from scratch.

3.3 The Role of Explicit Priors

Having established that SPT enables more reliable architectural evaluation and improves SSM performance, we now examine S4's design complexity. S4's theoretically-motivated design enables long-range signal propagation, explaining its slight advantage over SPT Transformers. While various S4 simplifications have been proposed, we show that SPT enables even simpler diagonal linear RNNs to match S4 performance.

For SSMs (see [Gu *et al.*, 2022] for details), given input scalar sequence u, a linear recurrence generates hidden state

Approach	SC		sCIFAR	BIDMC		
	Causal	Bi.		HR	RR	SpO2
S4	93.60	96.08	91.13	**0.999**†	**0.994**†	**0.999**†
Transformers	84.55	86.93	79.41	0.998	0.981	0.998
S4 + SPT	**95.09**	**96.52**	**91.67**	0.999	0.990	0.997
Transformers + SPT	86.13	91.49	90.29	0.992	0.956	0.993

Table 3: **Additional Experiments.** Performance on Speech Commands (SC), sCIFAR (accuracy) and BIDMC (R2) tasks. Results for trained from scratch S4 taken from [Gu *et al.*, 2022], except for BIDMC (denoted by †) that are reproduced for the more interpretable R2 score.

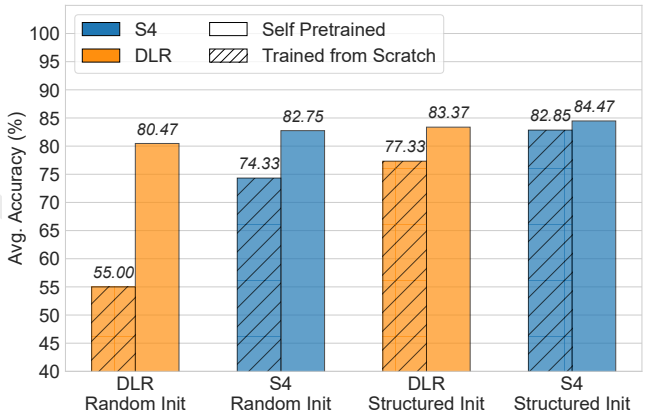

Figure 2: Average performance of models when trained from scratch or self pretrained, for different sets of initializations prior to pretraining. See the original paper [Amos *et al.*, 2024] for per-task results.

$\vec{\boldsymbol{x}}_n$ at timestep n and a scalar output sequence y as:

$$\begin{aligned} \vec{\boldsymbol{x}}_n &= \boldsymbol{A}\vec{\boldsymbol{x}}_{n-1} + \boldsymbol{B}u_n \quad & \boldsymbol{A} \in \mathbb{C}^{N\times N}, \boldsymbol{B} \in \mathbb{C}^{N\times 1} \\ y_n &= \boldsymbol{C}\vec{\boldsymbol{x}}_n & \boldsymbol{C} \in \mathbb{C}^{1\times N} \end{aligned} \tag{1}$$

This can be computed by convolving u with kernel defined by $K_k = \boldsymbol{C}^T \boldsymbol{A}^k \boldsymbol{B}$. S4 uses a specialized parameterization with transformations:

$$\boldsymbol{A} = \boldsymbol{\Lambda} - \boldsymbol{P}\boldsymbol{Q}^* \tag{2.1}$$

$$\bar{\boldsymbol{A}} = (\boldsymbol{I} - \Delta/2 \cdot \boldsymbol{A})^{-1}(\boldsymbol{I} + \Delta/2 \cdot \boldsymbol{A}) \tag{2.2}$$

$$\bar{\boldsymbol{B}} = (\boldsymbol{I} - \Delta/2 \cdot \boldsymbol{A})^{-1}\Delta\boldsymbol{B} \quad \bar{\boldsymbol{C}} = \boldsymbol{C} \tag{2.3}$$

$$\boldsymbol{K}_k = \bar{\boldsymbol{C}}^T \bar{\boldsymbol{A}}^k \bar{\boldsymbol{B}} \tag{2.4}$$

with learnable parameters $\boldsymbol{\Lambda}, \boldsymbol{P}, \boldsymbol{Q}, \boldsymbol{B}, \boldsymbol{C}, \Delta$ where $\boldsymbol{\Lambda} \in Diag(\mathbb{C}^{N\times N})$, $\boldsymbol{P}, \boldsymbol{Q} \in \mathbb{C}^{N\times 1}$. S4 uses principled initialization for slow kernel decay to capture long-range dependencies.

[Gupta *et al.*, 2022b] proposed a simplified Diagonal Linear RNN (DLR):

$$\begin{aligned} \vec{\boldsymbol{x}}_n &= \boldsymbol{\Lambda}\vec{\boldsymbol{x}}_{n-1} + \boldsymbol{1}u_n \quad & \boldsymbol{\Lambda} \in diag(\mathbb{C}^{N\times N}) \\ y_n &= \boldsymbol{C}\vec{\boldsymbol{x}}_n & \boldsymbol{C} \in \mathbb{C}^{1\times N} \end{aligned} \tag{3}$$

where $\boldsymbol{1}$ is the all-ones vector. DLR is computationally simpler than S4 yet reportedly matches state-of-the-art SSMs on token-level tasks. We investigate when S4's complex design can be replaced by DLR, testing both on the hardest LRA tasks (ListOps, Text, Image, PathX) with two initialization schemes: random and "structured" (designed for long-range dependencies).

Results in Figure 2 show that when trained from scratch, DLR lags behind S4 (77 vs 83), confirming that S4's specialized initialization and parameterization are critical. However, with SPT, DLR outperforms from-scratch S4 (83.4 vs 82.8) and approaches SPT S4 (83.4 vs 84.5) suggesting that the data-driven priors from pretraining can largely replace manual biases.

These findings have broader implications. First, this is the first demonstration of vanilla diagonal linear RNNs achieving competitive LRA performance without normalization or specialized initialization [Orvieto *et al.*, 2023]. Second, many global convolution designs follow similar principles to SSMs, like generating smooth decaying kernels [Li *et al.*, 2023] or applying deterministic transformations [Fu *et al.*, 2023]. Our results suggest these explicit design steps become less critical with self-pretraining.

3.4 Additional Experiments

We tested SPT on three additional datasets across different modalities:

- **Speech Commands (SC):** Raw speech waveforms (16K length) for 35-way classification [Warden, 2018], testing both causal and bidirectional models.
- **sCIFAR:** Sequential CIFAR-10 with RGB channels as features (richer than LRA's grayscale Image task).
- **BIDMC:** Three regression tasks predicting health metrics (RR, HR, SpO2) from 4K-length physiological signals.

Results in Table 3 further support our claims. On SC and sCIFAR, SPT significantly improves Transformer performance while modestly enhancing S4, substantially narrowing their performance gap. For SC, Transformers gain 5% with SPT, and the gap between causal and bidirectional S4 variants diminishes, similar to our LRA observations. On sCIFAR, Transformers improve by 11% with SPT, nearly matching S4 (90.3 vs 91.7). BIDMC results show minimal gains as both models already achieve near-perfect performance. These findings suggest that performance underestimation likely affects other domains where training from scratch remains standard practice [Delétang *et al.*, 2023; Velickovic *et al.*, 2022; Dwivedi *et al.*, 2022].

References

[Amos *et al.*, 2024] Ido Amos, Jonathan Berant, and Ankit Gupta. Never train from scratch: Fair comparison of long-

sequence models requires data-driven priors. In *Proc. of ICLR*, 2024.

[Baevski *et al.*, 2020] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. of NeurIPS*, 2020.

[Dao *et al.*, 2022] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Proc. of NeurIPS*, 2022.

[Delétang *et al.*, 2023] Grégoire Delétang, Anian Ruoss, Jordi Grau-Moya, Tim Genewein, Li Kevin Wenliang, Elliot Catt, Chris Cundy, Marcus Hutter, Shane Legg, Joel Veness, and Pedro A. Ortega. Neural networks and the chomsky hierarchy. In *Proc. of ICLR*, 2023.

[Diao and Loynd, 2023] Cameron Diao and Ricky Loynd. Relational attention: Generalizing transformers for graph-structured tasks. In *Proc. of ICLR*, 2023.

[Dwivedi *et al.*, 2022] Vijay Prakash Dwivedi, Ladislav Rampásek, Michael Galkin, Ali Parviz, Guy Wolf, Anh Tuan Luu, and Dominique Beaini. Long range graph benchmark. In *Proc. of NeurIPS*, 2022.

[El-Nouby *et al.*, 2021] Alaaeldin El-Nouby, Gautier Izacard, Hugo Touvron, Ivan Laptev, Hervé Jegou, and Edouard Grave. Are large-scale datasets necessary for self-supervised pre-training? *ArXiv preprint*, 2021.

[Fu *et al.*, 2023] Daniel Y. Fu, Elliot L. Epstein, Eric Nguyen, Armin W. Thomas, Michael Zhang, Tri Dao, Atri Rudra, and Christopher Ré. Simple hardware-efficient long convolutions for sequence modeling. In *Proc. of ICML*, Proceedings of Machine Learning Research, 2023.

[Gu *et al.*, 2020] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. In *Proc. of NeurIPS*, 2020.

[Gu *et al.*, 2022] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *Proc. of ICLR*, 2022.

[Gupta *et al.*, 2022a] Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. In *Proc. of NeurIPS*, 2022.

[Gupta *et al.*, 2022b] Ankit Gupta, Harsh Mehta, and Jonathan Berant. Simplifying and understanding state space models with diagonal linear rnns. *ArXiv preprint*, 2022.

[He *et al.*, 2022] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, 2022.

[Jumper *et al.*, 2021] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, (7873), 2021.

[Kim *et al.*, 2020] Junkyung Kim, Drew Linsley, Kalpit Thakkar, and Thomas Serre. Disentangling neural mechanisms for perceptual grouping. In *Proc. of ICLR*, 2020.

[Krishna *et al.*, 2023] Kundan Krishna, Saurabh Garg, Jeffrey Bigham, and Zachary Lipton. Downstream datasets make surprisingly good pretraining corpora. In *Proc. of ACL*, 2023.

[Li *et al.*, 2023] Yuhong Li, Tianle Cai, Yi Zhang, Deming Chen, and Debadeepta Dey. What makes convolutional models great on long sequence modeling? In *Proc. of ICLR*, 2023.

[Linsley *et al.*, 2018] Drew Linsley, Junkyung Kim, Vijay Veerabadran, Charles Windolf, and Thomas Serre. Learning long-range spatial dependencies with horizontal gated recurrent units. In *Proc. of NeurIPS*, 2018.

[Liu *et al.*, 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*, 2019.

[Ma *et al.*, 2023] Xuezhe Ma, Chunting Zhou, Xiang Kong, Junxian He, Liangke Gui, Graham Neubig, Jonathan May, and Luke Zettlemoyer. Mega: Moving average equipped gated attention. In *Proc. of ICLR*, 2023.

[Maas *et al.*, 2011] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proc. of ACL*, 2011.

[Nangia and Bowman, 2018] Nikita Nangia and Samuel Bowman. ListOps: A diagnostic dataset for latent tree learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, 2018.

[Nguyen *et al.*, 2022] Eric Nguyen, Karan Goel, Albert Gu, Gordon W. Downs, Preey Shah, Tri Dao, Stephen Baccus, and Christopher Ré. S4ND: modeling images and videos as multidimensional signals with state spaces. In *Proc. of NeurIPS*, 2022.

[Orvieto *et al.*, 2023] Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Çaglar Gülçehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *Proc. of ICML*, Proceedings of Machine Learning Research, 2023.

[Radev *et al.*, 2009] Dragomir R. Radev, Pradeep Muthukrishnan, and Vahed Qazvinian. The ACL Anthology network corpus. In *Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries (NLPIR4DL)*, 2009.

[Raffel *et al.*, 2020] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 2020.

[Reed *et al.*, 2022] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley D. Edwards, Nicolas Manfred Otto Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent. *Trans. Mach. Learn. Res.*, 2022.

[Shirzad *et al.*, 2023] Hamed Shirzad, Ameya Velingker, Balaji Venkatachalam, Danica J. Sutherland, and Ali Kemal Sinop. Exphormer: Sparse transformers for graphs. In *Proc. of ICML*, Proceedings of Machine Learning Research, 2023.

[Su *et al.*, 2021] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *ArXiv preprint*, 2021.

[Tay *et al.*, 2021] Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena : A benchmark for efficient transformers. In *Proc. of ICLR*, 2021.

[Touvron *et al.*, 2023] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, 2023.

[Velickovic *et al.*, 2022] Petar Velickovic, Adrià Puigdomènech Badia, David Budden, Razvan Pascanu, Andrea Banino, Misha Dashevskiy, Raia Hadsell, and Charles Blundell. The CLRS algorithmic reasoning benchmark. In *Proc. of ICML*, Proceedings of Machine Learning Research, 2022.

[Warden, 2018] Pete Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *ArXiv preprint*, 2018.

[Zuo *et al.*, 2022] Simiao Zuo, Xiaodong Liu, Jian Jiao, Denis Charles, Eren Manavoglu, Tuo Zhao, and Jianfeng Gao. Efficient long sequence modeling via state space augmented transformer. *ArXiv preprint*, 2022.