

# White-Box Deep Network Architectures via Compression and Optimization

Druv Pai

UC Berkeley



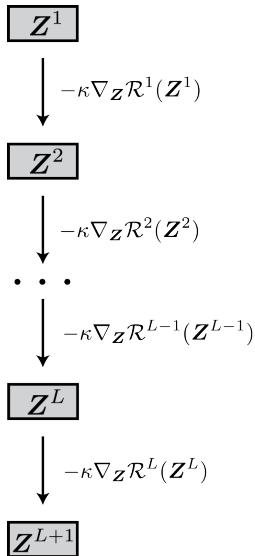
## Lectures so Far

- **History** of the pursuit and study of intelligence.
- **Learning and sampling** via analytic methods, denoising, and/or compression.
- **Objectives for representation learning**: information gain.
- **Deep neural networks** via information gain optimization.

**This lecture: novel deep architectures!**

Use *prescriptive theory* to build *novel architectures* with empirical benefits. Plus, open problems!

## Recall: Unrolling for CRATE



- **Unrolled optimization:** each layer conducts an optimization step, stylized as

$$Z^{\ell+1} \leftarrow Z^{\ell} - \kappa \nabla_Z \mathcal{R}^{\ell}(Z^{\ell})$$

on the **sparse rate reduction** objective.

- Two step procedure:
  - **Compression:**

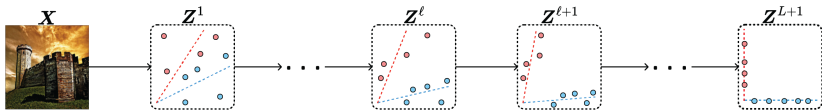
$$Z^{\ell+1/2} \approx Z^{\ell} - \kappa \nabla_Z R^c(Z^{\ell} \mid U_{[K]}^{\ell})$$

- **Sparsification:**

$$\begin{aligned} Z^{\ell+1} &\approx \arg \max_{Z: Z^{\ell+1/2} \approx D^{\ell} Z} \{R(Z) - \lambda \|Z\|_1\} \\ &\approx \arg \min_Z \left\{ \frac{1}{2} \|Z^{\ell+1/2} - D^{\ell} Z\|_2^2 + \lambda' \|Z\|_1 \right\} \end{aligned}$$

# Causal CRATE?

Question: How to incrementally optimize the **sparse rate reduction** *one token at a time*?



$$\text{SRR}(\mathbf{Z} \mid \mathbf{U}_{[K]}) := R(\mathbf{Z}) - R^c(\mathbf{Z} \mid \mathbf{U}_{[K]}) - \lambda \|\mathbf{Z}\|_1$$

# Causal CRATE

- **Causal unrolled optimization:** each layer conducts an optimization step for each token  $t$ , stylized as

$$\mathbf{z}_t^{\ell+1} \leftarrow \mathbf{z}_t^\ell - \kappa \nabla_{\mathbf{z}_t} \mathcal{R}_t^\ell(\mathbf{Z}_{:t}^\ell),$$

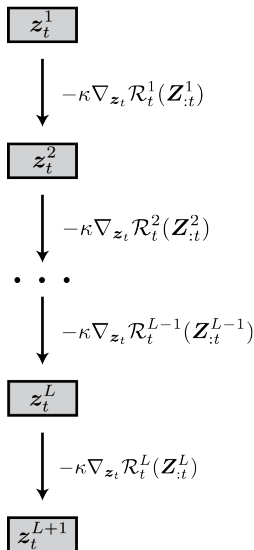
where  $\mathbf{Z}_{:t}^\ell := [\mathbf{z}_1^\ell, \dots, \mathbf{z}_t^\ell]$ .

- Two step procedure for each token  $t$ :
  - **Compression:**

$$\mathbf{z}_t^{\ell+1/2} \approx \mathbf{z}_t^\ell - \kappa \nabla_{\mathbf{z}_t} R^c(\mathbf{Z}_{:t}^\ell \mid \mathbf{U}_{[K]}^\ell)$$

- **Sparsification:**

$$\mathbf{z}_t^{\ell+1} \approx \arg \min_{\mathbf{z}_t} \left\{ \frac{1}{2} \|\mathbf{z}_t^{\ell+1/2} - \mathbf{D}^\ell \mathbf{z}_t\|_2^2 + \lambda' \|\mathbf{z}_t\|_1 \right\}$$



## Causal CRATE: Compression Operator

If  $(U_k)_{k=1}^K \approx \text{orthogonal} + \text{p/w} \approx \text{orthogonal} + \approx \text{support } \mathbf{Z}$ :

$$\begin{aligned}\nabla_{\mathbf{z}_t} R^c(\mathbf{Z}_{:t} \mid \mathbf{U}_{[K]}) &= [\nabla_{\mathbf{Z}_{:t}} R^c(\mathbf{Z}_{:t} \mid \mathbf{U}_{[K]})] \mathbf{e}_t \\ &\approx \beta \left[ \mathbf{Z}_{:t} - \beta \sum_{k=1}^K \mathbf{U}_k (\mathbf{U}_k^\top \mathbf{Z}_{:t}) (\mathbf{U}_k^\top \mathbf{Z}_{:t})^\top (\mathbf{U}_k^\top \mathbf{Z}_{:t}) \right] \mathbf{e}_t \\ &= \beta \left[ \mathbf{z}_t - \beta \sum_{k=1}^K \mathbf{U}_k (\mathbf{U}_k^\top \mathbf{Z}_{:t}) (\mathbf{U}_k^\top \mathbf{Z}_{:t})^\top (\mathbf{U}_k^\top \mathbf{z}_t) \right]\end{aligned}$$

Gradient shaping/"non-parametric autoregression":

$$\nabla_{\mathbf{z}_t} R^c(\mathbf{Z}_{:t} \mid \mathbf{U}_{[K]}) \approx \beta \left[ \mathbf{z}_t - \beta \sum_{k=1}^K \mathbf{U}_k (\mathbf{U}_k^\top \mathbf{Z}_{:t}) \text{softmax} \left\{ (\mathbf{U}_k^\top \mathbf{Z}_{:t})^\top (\mathbf{U}_k^\top \mathbf{z}_t) \right\} \right]$$

# Causal MSSA

$$\nabla_{\mathbf{z}_t} R^c(\mathbf{Z}_{:t} \mid \mathbf{U}_{[K]}) \approx \beta \left[ \mathbf{z}_t - \beta \sum_{k=1}^K \mathbf{U}_k (\mathbf{U}_k^\top \mathbf{Z}_{:t}) \text{softmax} \left\{ (\mathbf{U}_k^\top \mathbf{Z}_{:t})^\top (\mathbf{U}_k^\top \mathbf{z}_t) \right\} \right]$$

Causal Multi-head Subspace Self-Attention:

$$\text{MSSA}(\mathbf{z}_t \mid \mathbf{U}_{[K]}, \mathbf{Z}_{:t-1}) := \beta [\mathbf{U}_1, \dots, \mathbf{U}_K] \begin{bmatrix} (\mathbf{U}_1^\top \mathbf{Z}_{:t}) \text{softmax} \{ (\mathbf{U}_1^\top \mathbf{Z}_{:t})^\top (\mathbf{U}_1^\top \mathbf{z}_t) \} \\ \vdots \\ (\mathbf{U}_K^\top \mathbf{Z}_{:t}) \text{softmax} \{ (\mathbf{U}_K^\top \mathbf{Z}_{:t})^\top (\mathbf{U}_K^\top \mathbf{z}_t) \} \end{bmatrix}$$

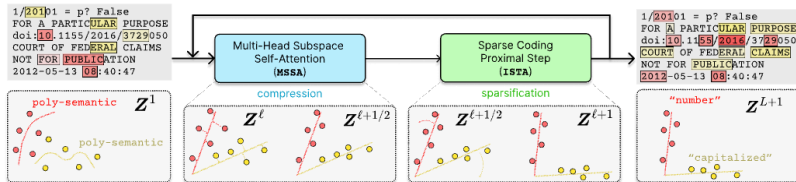
where  $\mathbf{Z}_{:t} = [\mathbf{Z}_{:t-1}, \mathbf{z}_t]$ .

$$\mathbf{z}_t^{\ell+1/2} := \underbrace{(1 - \beta\kappa) \mathbf{z}_t^\ell}_{\text{residual}} + \underbrace{\beta\kappa \text{MSSA}(\mathbf{z}_t^\ell \mid \mathbf{U}_{[K]}^\ell, \mathbf{Z}_{:t-1}^\ell)}_{\text{causal attention-like w/ cache}}$$

# Causal CRATE: Sparsification Operator

Exactly the same as CRATE!

$$z_t^{\ell+1} = \text{ISTA}(z_t^{\ell+1/2} \mid D^\ell).$$





# Causal CRATE Experimental Results

Validation cross-entropy loss after pre-training a LLM:

	#parameters	OWT	LAMBADA	WikiText	PTB	Avg
GPT2-Base	124M	2.85	4.12	3.89	4.63	3.87
GPT2-Small	64M	3.04	4.49	4.31	5.15	4.25
CRATE-GPT2-Base	60M	3.37	4.91	4.61	5.53	4.61

Interpretability scores:

	Mean (↑, darker green means more interpretable)						Variance (↓, darker red means less steady)					
	Top-and-Random		Random-only		Anthropic		Top-and-Random		Random-only		Anthropic	
	CRATE	GPT-2	CRATE	GPT-2	CRATE	GPT-2	CRATE	GPT-2	CRATE	GPT-2	CRATE	GPT-2
1L	3.9	8.8	4.8	8.9	10.1	14.2	0.0	0.0	0.0	0.0	0.0	0.0
2L	8.05	4.2	6.95	1.95	11.35	10.2	0.06	0.01	1.1	0.12	0.0	0.25
3L	9.1	3.57	8.43	1.37	11.23	9.2	0.26	7.51	1.2	1.93	1.14	19.21
6L	7.96	5.4	6.36	3.14	10.4	8.52	2.29	20.85	1.87	18.39	2.01	32.56
12L	6.8	6.34	5.12	2.67	8.88	8.65	7.09	11.35	2.83	7.48	18.3	24.65

Causal CRATE achieves *close* performance and *superior interpretability* at *similar compute costs*.

The **best performance** of all similar mathematically derived architectures **by far!**

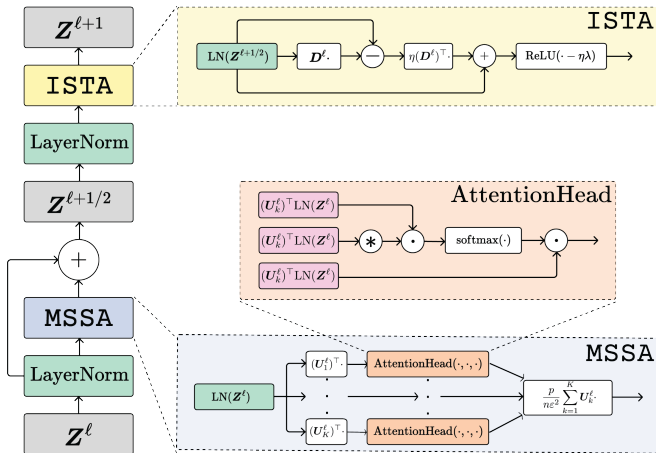
## Causal CRATE: Open Problems and Future Work

The performance lags in *text* more than *vision*.

What is a *sparse rate reduction*-type objective and optimization strategy for *time-series data* (e.g., text/video)?

- Our best guess: Objective should also encourage  $(z_t)_{t=1}^N$  to have organized *dynamics*.
- Work is ongoing on this.

# Improving Scaling Laws in CRATE: CRATE- $\alpha$



**Major bottleneck:** Dictionary  $D^\ell$  is *square* (complete).

**Why not make it *overcomplete* (wide)?**

## Sparsification Block in CRATE- $\alpha$

- Overcomplete dictionary learning block: two iterations of ISTA, initialized at  $\mathbf{0}$ , to get  $\mathbf{Z}^{\ell+1/2}$ .
- Sparse codes are now larger than inputs, so *take the denoised inputs* as features.

Provides the **Orthogonal Dictionary Learning (ODL)** block:

$$\text{ODL}(\mathbf{Z}^{\ell+1/2}) = \mathbf{D}^\ell \text{ISTA}(\text{ISTA}(\mathbf{0} \mid \mathbf{Z}^{\ell+1/2}, \mathbf{D}^\ell) \mid \mathbf{Z}^{\ell+1/2}, \mathbf{D}^\ell)$$

where

$$\text{ISTA}(\mathbf{A} \mid \mathbf{Z}, \mathbf{D}) = \text{ReLU}(\mathbf{A} - \mathbf{D}^\top (\mathbf{Z} - \mathbf{D}\mathbf{A}) - \kappa\lambda\mathbf{1})$$

**MSSA + ODL = CRATE- $\alpha$ !**

# CRATE- $\alpha$ Experimental Results

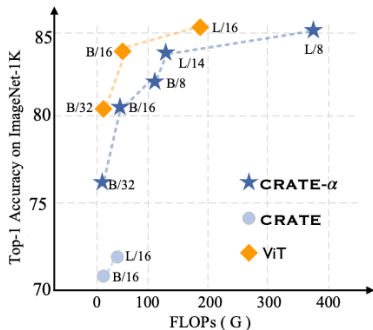
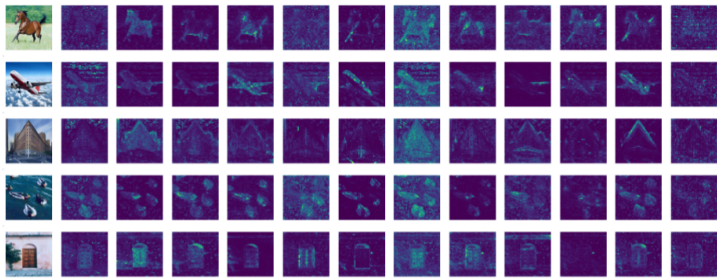


Figure 5: Visualization of segmentation on COCO val2017 [20] with MaskCut [43]. (Top row)

Table 4: The comparison between CRATE and CRATE- $\alpha$  on the NLP task using the OpenWebText dataset.

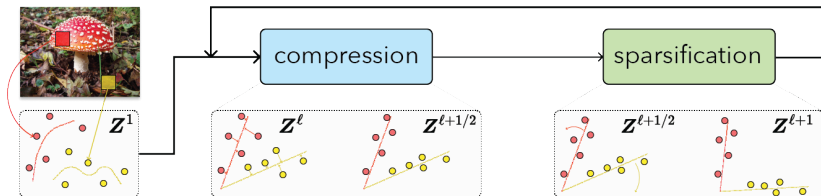
	GPT-2-base	CRATE-base	CRATE- $\alpha$ -small	CRATE- $\alpha$ -base
Model size	124M	60M	57M	120M
CE val loss	2.85	3.37	3.28	3.14



# ToST: Linear-Time White-Box Architecture

**Motivation:** For large #s of tokens, *quadratic attention* too expensive to train on.

**Idea:** Use a *different* **compression measure**  $R^c$  to get a *different* **attention mechanism**.



Previous compression measure *explicitly parameterized* low-rank Gaussian mixture model covariances by  $\mathbf{U}_{[K]}$  to group tokens.

# Token Clustering Model

## Cluster model:

- Use a soft assignment  $\mathbf{\Pi} = [\boldsymbol{\pi}_1, \dots, \boldsymbol{\pi}_K] \in [0, 1]^{n \times K}$  of samples to clusters
- $\mathbf{\Pi} \mathbf{1}_K = \mathbf{1}_n \implies$  rows are probability vectors

**Rate reduction:**  $\Delta R(\mathbf{Z} \mid \mathbf{\Pi}) := R(\mathbf{Z}) - R^c(\mathbf{Z} \mid \mathbf{\Pi})$  where

$$R^c(\mathbf{Z} \mid \mathbf{\Pi}) := \frac{1}{2} \sum_{k=1}^K \frac{\langle \boldsymbol{\pi}_k, \mathbf{1} \rangle}{n} \log \det \left( \mathbf{I}_d + \frac{d}{\varepsilon^2} \cdot \frac{\mathbf{Z} \operatorname{diag}(\boldsymbol{\pi}_k) \mathbf{Z}^\top}{\langle \boldsymbol{\pi}_k, \mathbf{1} \rangle} \right)$$

*Note:* reduces to supervised  $R^c$  if  $\mathbf{\Pi} \in \{0, 1\}^{n \times K}$ .

# Variational Form

Low-rank structure still enters naturally into the picture:

**Theorem ([Wu+24], Corollary 1, Simplified):** For *orthogonal*  $U_{[K]} \subseteq (\mathbb{R}^{d \times p})^K$  supporting  $\mathbf{Z}$ :

$$R^c(\mathbf{Z}, \mathbf{\Pi}) \leq R_{\text{var}}^c(\mathbf{Z}, \mathbf{\Pi} \mid U_{[K]})$$

where

$$R_{\text{var}}^c(\mathbf{Z}, \mathbf{\Pi} \mid U_{[K]}) := \frac{1}{2} \sum_{k=1}^K \frac{\langle \boldsymbol{\pi}_k, \mathbf{1} \rangle}{n} \sum_{i=1}^d \log \left( 1 + \frac{d}{\varepsilon^2} \cdot \frac{[(U_k^\top \mathbf{Z}) \text{diag}(\boldsymbol{\pi}_k)(U_k^\top \mathbf{Z})^\top]_{ii}}{\langle \boldsymbol{\pi}_k, \mathbf{1} \rangle} \right)$$

This is a corollary of a more general theorem from paper.

**Important:**  $R_{\text{var}}^c$  uses *just* log, *not* logdet!



## Gradient of Variational Form

$$\nabla_{\mathbf{Z}} R_{\text{var}}^c(\mathbf{Z}, \mathbf{\Pi} \mid \mathbf{U}_{[K]}) = \frac{1}{n} \sum_{k=1}^K \mathbf{U}_k \mathbf{D}(\mathbf{Z}, \boldsymbol{\pi}_k \mid \mathbf{U}_k) \mathbf{U}_k^\top \mathbf{Z} \text{diag}(\boldsymbol{\pi}_k)$$

where<sup>1</sup>

$$\mathbf{D}(\mathbf{Z}, \boldsymbol{\pi}_k \mid \mathbf{U}_k) := \frac{d}{\varepsilon^2} \text{diag} \left( \left[ \mathbf{1} + \frac{d}{\varepsilon^2} (\mathbf{U}_k^\top \mathbf{Z})^{\odot 2} \frac{\boldsymbol{\pi}_k}{\langle \boldsymbol{\pi}_k, \mathbf{1} \rangle} \right]^{\odot -1} \right)$$

Minimize the upper bound:

$$\mathbf{Z}^{\ell+1/2} := \mathbf{Z}^\ell - \kappa \nabla_{\mathbf{Z}} R_{\text{var}}^c(\mathbf{Z}^\ell, \mathbf{\Pi}^\ell \mid \mathbf{U}_{[K]}^\ell)$$

---

<sup>1</sup>In the formula,  $\odot$  = elementwise (e.g., squaring, inverse).

## Estimating Cluster Membership

Last step: estimating  $\Pi$  from  $Z$ .

$$\Pi(Z) = \begin{bmatrix} \text{softmax}(\frac{1}{\tau} [\|U_1^\top z^1\|_2^2, \dots, \|U_K^\top z^1\|_2^2]) \\ \vdots \\ \text{softmax}(\frac{1}{\tau} [\|U_1^\top z^n\|_2^2, \dots, \|U_K^\top z^n\|_2^2]) \end{bmatrix} \in [0, 1]^{n \times K}$$

**Token Statistics Self-Attention (TSSA):**

$$\text{TSSA}(Z) := -\frac{\kappa}{n} \sum_{k=1}^K U_k D_k(Z, \pi_k(Z)) U_k^\top Z \text{diag}(\pi_k(Z))$$

$$Z^{\ell+1/2} := Z^\ell + \text{TSSA}(Z^\ell).$$

# ToST Architecture

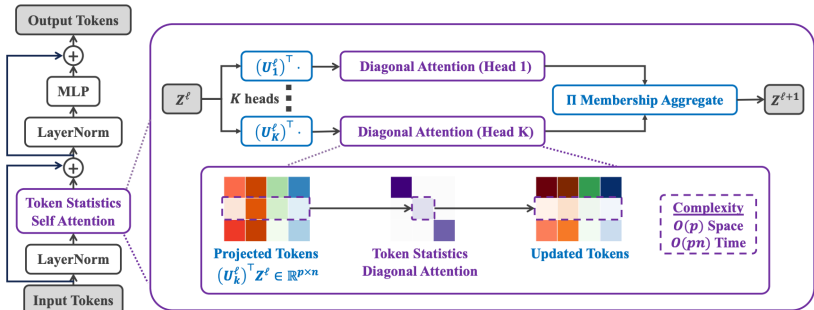


Table 1: Time and space complexity for attention operators in different transformer architectures: ViT (Dosovitskiy et al., 2020), CRATE (Yu et al., 2024), XCiT (Ali et al., 2021), and the proposed ToST.

	ViT	CRATE	XCiT	ToST (ours)
Compute time complexity	$\mathcal{O}(pn^2)$	$\mathcal{O}(pn^2)$	$\mathcal{O}(p^2n)$	$\mathcal{O}(pn)$
Memory complexity	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$	$\mathcal{O}(p^2)$	$\mathcal{O}(p)$

# ToST Results

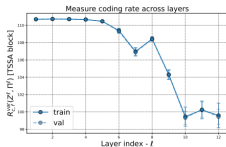
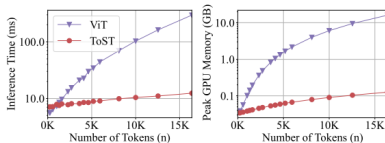


Table 3: Long-Range Arena (LRA) performance comparison.

Model	ListOps	Text	Retrieval	Image	Pathfinder	Avg
Reformer	<b>37.27</b>	56.10	53.40	38.07	68.50	50.56
BigBird	36.05	64.02	59.29	40.83	74.87	54.17
LinFormer	16.13	<u>65.90</u>	53.09	42.34	<u>75.30</u>	50.46
Performer	18.01	<u>65.40</u>	53.82	42.77	<b>77.05</b>	51.18
Transformer	37.11	65.21	<u>79.14</u>	<u>42.94</u>	71.83	<u>59.24</u>
<b>ToST (ours)</b>	<u>37.25</u>	<b>66.75</b>	<b>79.46</b>	<b>46.62</b>	69.41	<b>59.90</b>

Datasets	ToST-T(iny)	ToST-S(mall)	ToST-M(edium)	XCiT-S	XCiT-M	ViT-S	ViT-B(ase)
# parameters	5.8M	22.6M	68.1M	24.9M	80.2M	22.1M	86.6 M
ImageNet	67.3	77.9	80.3	80.5	81.5	79.8	81.8
ImageNet RealL	72.2	84.1	85.6	85.6	85.9	85.6	86.7
CIFAR10	95.5	96.5	97.5	98.1	98.3	98.6	98.8
CIFAR100	78.3	82.7	84.5	86.1	87.6	88.8	89.3
Oxford Flowers-102	88.6	92.8	94.2	93.9	94.0	94.0	95.7
Oxford-IIIT-Pets	85.6	91.1	92.8	92.9	94.0	92.8	94.1

Model	# params	OWT	Lambada	Wikitext	PTB	Avg ↓
GPT2-Base	124M	2.84	4.32	4.13	5.75	4.26
ToST-Base	110M	3.20	4.98	4.77	6.39	4.84
ToST-Medium	304M	2.88	4.45	4.30	5.64	4.32
ToST-Large	655M	2.72	4.32	3.99	5.03	4.02



## Summary

- Different unrolling schemes, different information gain, different optimization:  $\implies$  different architectures.
- Even when a novel architecture is very different from existing architectures, it can *still have good performance!*

## Open questions:

- How can we leverage advances in convex and nonconvex optimization to derive better architectures?
- How to build architectures for different problems where more bespoke representations are needed?
- How to build architectures which solve current challenges, e.g., efficiency, continual learning, interpretability, etc?

## References I

- [Bae+22] Christina Baek, Ziyang Wu, Kwan Ho Ryan Chan, et al. “Efficient maximal coding rate reduction by variational forms”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 500–508.
- [BM24] Hao Bai and Yi Ma. “Improving neuron-level interpretability with white-box language models”. In: *arXiv preprint arXiv:2410.16443* (2024).
- [Wu+24] Ziyang Wu, Tianjiao Ding, Yifu Lu, et al. “Token statistics transformer: Linear-time attention via variational rate reduction”. In: *arXiv preprint arXiv:2412.17810* (2024).

## References II

- [Yan+24] Jinrui Yang, Xianhang Li, Druv Pai, et al. “Scaling white-box transformers for vision”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 36995–37019.
- [Yu+23] Yaodong Yu, Sam Buchanan, Druv Pai, et al. “White-box transformers via sparse rate reduction”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 9422–9457.