AI in the Sciences and Engineering 2024: Lecture 18

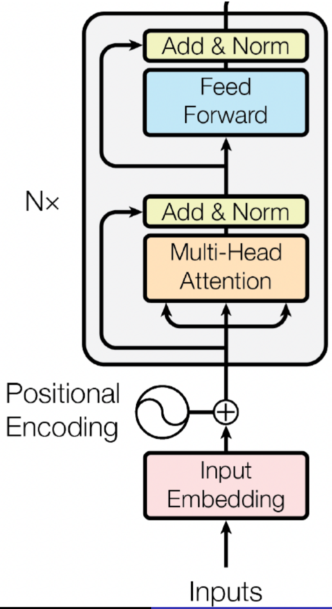
Siddhartha Mishra

Seminar for Applied Mathematics (SAM), D-MATH (and),
ETH AI Center,
ETH Zürich, Switzerland.

What you learnt so far

- Operator learning: Given Abstract PDE: $\mathcal{D}_a(u) = f$
- Learn Solution Operator: $\mathcal{G} : \mathcal{X} \mapsto \mathcal{Y}$ with $\mathcal{G}(a, f) = u$
- Approximate with Operator Learning Algorithms:
    - CNN/UNet
    - DeepONet
    - FNO
    - CNO
- We focus on Transformers

Final version of a Transformer Block



Caveat: Computational Complexity

- Computational Cost is Quadratic in # (Tokens) !!

$$\text{Compute} \sim \mathcal{O}(mnK^2)$$

- With K- Input Length, n Input features and m hidden dimension.
- But lots of possible Parallelism in Computation
- $\mathcal{O}(1)$ sequential operations.
- $\mathcal{O}(1)$ Path Length.
- Nevertheless, Infeasible for 2 or 3-d inputs.

Possible Solution

- Vision Transformers (ViT) of Dosovitskiy et. al.
- For $D \subset \mathbb{R}^2$ + input $v \in C(D, \mathbb{R}^C)$
- A sequence of Operators of the form:
- Patch Embeddings+ Positional Encoding: $\hat{v} = \hat{E}(v) + E_{pos}(v)$
- LayerNorm + MSA+ Residual: $\bar{u} = \hat{v} + MSA(LN(v))$
- LayerNorm + MLP+ Residual: $u = \bar{u} + MLP(LN(\bar{u}))$

Computational Complexity

- Given an Image at resolution $H \times W$
- Standard Transformer needs

$$\text{Compute} \sim \mathcal{O}((HW)^2)$$

- ViT needs

$$\text{Compute} \sim \mathcal{O}\left(\frac{(HW)^2}{p^4}\right)$$

- Still not scalable for small patch size p

Another Idea: Windowed Attention

- Introduced in Liu et. al.
- Use Windowed Attention:

- With M-Windows,

$$\text{Compute} \sim \mathcal{O}\left(\frac{HWM^2}{p^2}\right)$$

Operator version of Windowed Attention

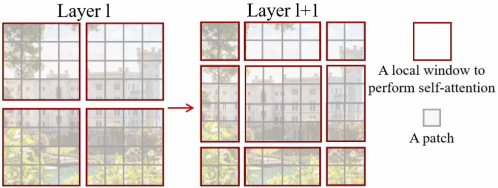
- For layer ℓ, Assume $D = \cup_{q=1}^{M} D_q^{w,\ell}$
- With non-overlapping Windows.
- Windowed Attention is instantiated as Operator:

$$\mathrm{u}(x) = \mathbb{A}_W(\mathrm{v})(x) = \mathrm{W} \int\limits_{D_{q_x}^{w,\ell}} \frac{e^{\frac{\langle \mathrm{Qv}(x), \mathrm{Kv}(y) \rangle}{\sqrt{m}}}}{\int\limits_{D_{q_x}^{w,\ell}} e^{\frac{\langle \mathrm{Qv}(z), \mathrm{Kv}(y) \rangle}{\sqrt{m}}} dz} \mathrm{Vv}(y) dy.$$

- Where $1 \leq q_x \leq M$ such that $x \in D_{q_x}^{w,\ell}$

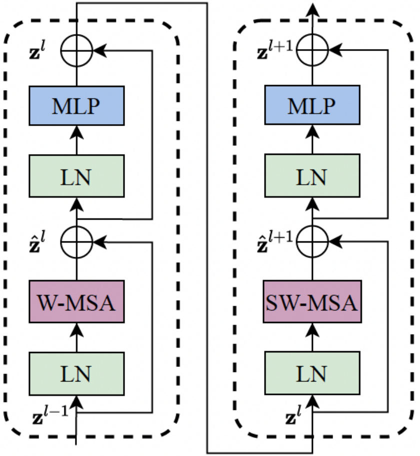
Shifting the Windows

- How to Tokens outside the Window ?
- Solution: Window shifts across Layers !!

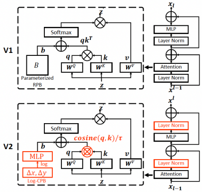


- Reduces Path Length across Tokens.

Swin Transformer Block

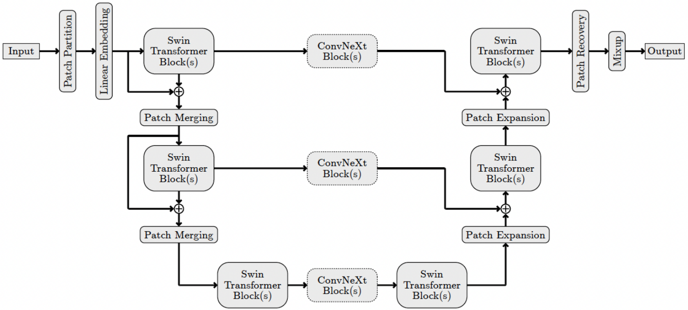


Modifications for Scalability

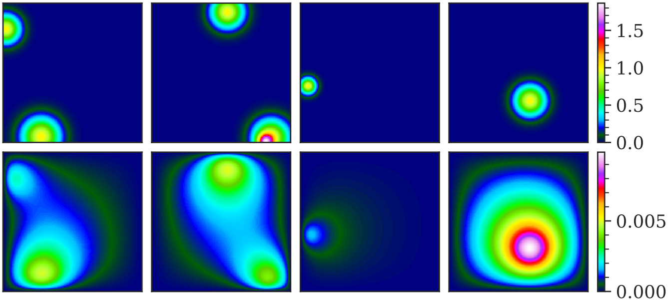


- Replace scaled dot product with Scaled Cosine
- Use MLPs on Relative Position Coordinates to generate positional encodings.

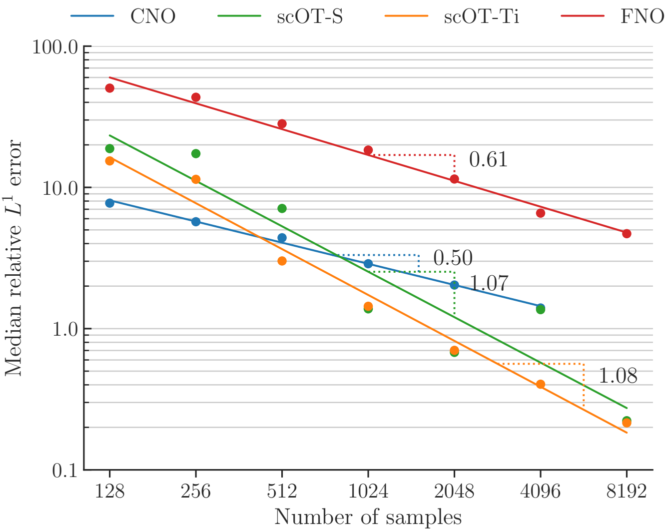
scOT: scalable Operator Transformer



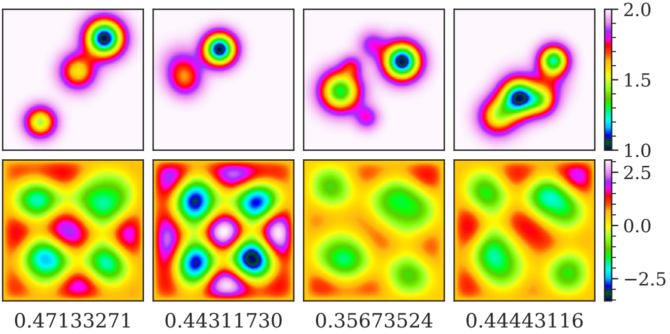
Poisson with Gaussian Sources



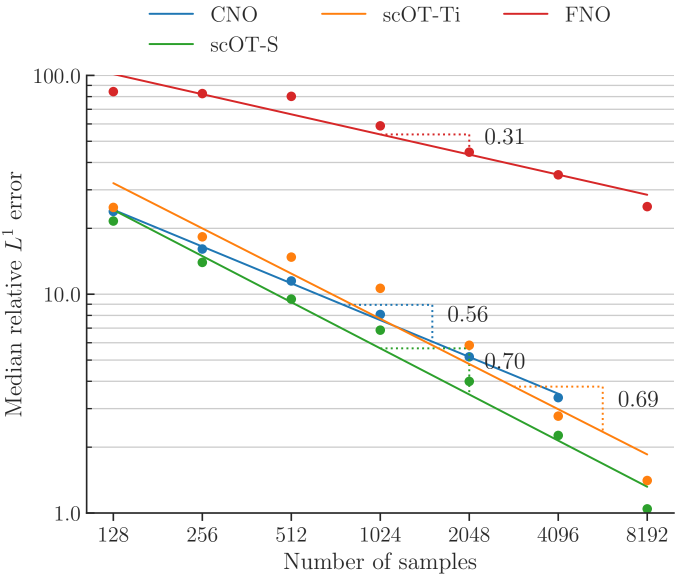
Poisson with Gaussian Sources



Helmholtz

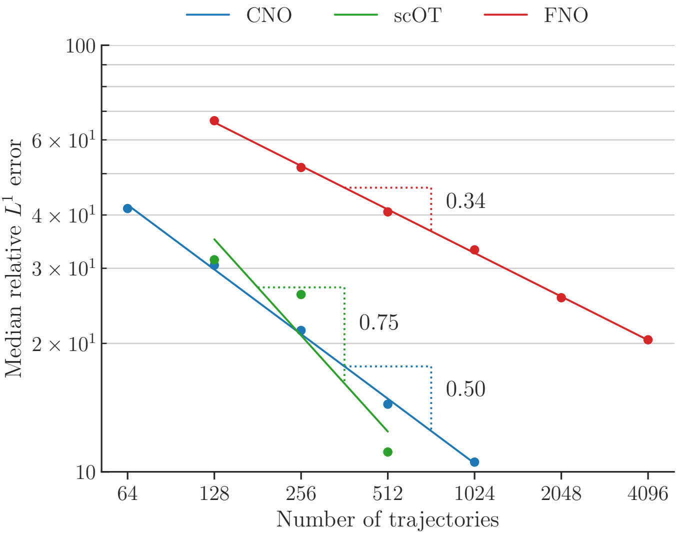


Helmholtz

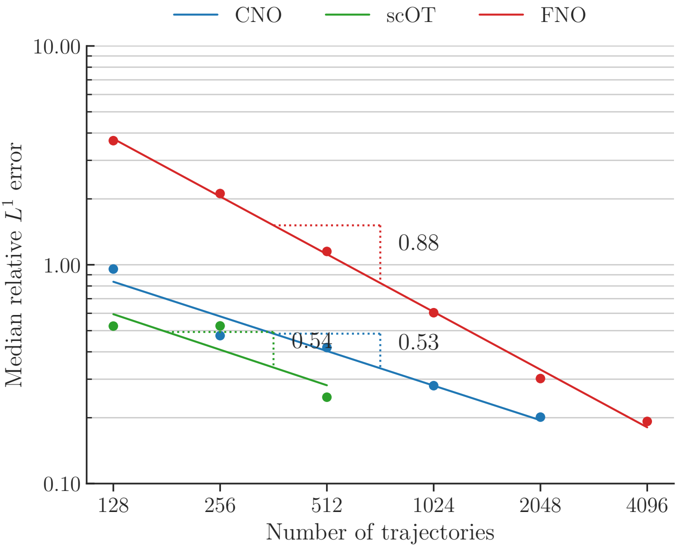


CNO
scOT-Ti
FNO
scOT-S
100.0
10.0
1.0
Median relative L^1 error
128
256
512
1024
2048
4096
8192
Number of samples
0.31
0.56
0.70
0.69

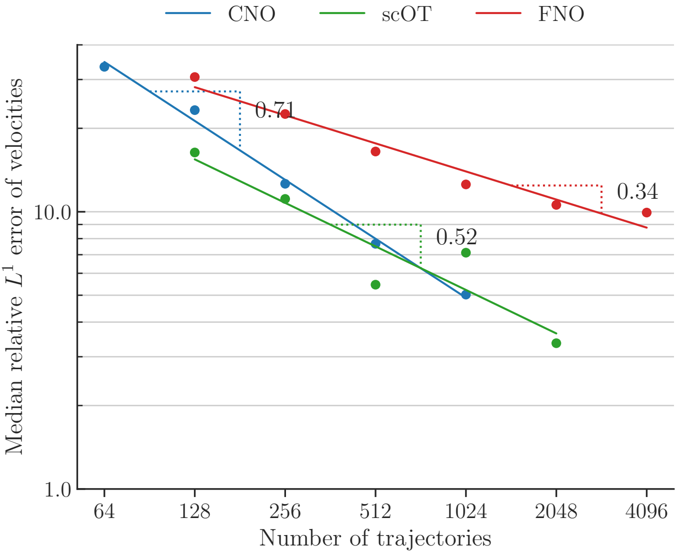
Wave Equation



Allen-Cahn Equation



Navier-Stokes



Possible Solutions for Sample Efficiency

- Add Physics:
  - PINN type residual based loss functions
  - Preconditioned Physics-informed ReNOs
- Use data better through Foundation Models

Navier-Stokes