Equivariant transformer is all you need

Akio Tomiya (IPUT Osaka, Assistant Prof.)

+ Yuki Nagai (JAEA, Senior Scientist)
1. What is Machine learning?
2. Transformer and Attention
3. Target system
4. Equivariant Attention
5. SLMC (self-learning MC)
6. Results
Short summary
Towards to simulate LQCD: equivariant attention

• We propose **Attention blocks for physical systems!**
  • Machine learning for physics (Monte-Carlo + neural net approx)
  • It keeps field rotation/translation symmetry (equivariant)
  • It can capture *non-local correlation* while CNN-type is hard to do

• We perform self-learning Monte-Carlo with the attention for “O(3) Yukawa system” system in condensed matter
  • *Not for gauge system*. Only for global symmetry

• We find that **the attention layers improve acceptance rates systematically** for increasing the number layers
  • It shows scaling behavior as in large language models

(c.f. We have proposed **gauge symmetric convolution** applied on 4d Full QCD, see arXiv: 2103.11965)
What is machine learning?
Symmetry?
What is machine learning?

E.g. Linear regression (supervised learning)

Data: \( D = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots\} \)

\[
f_{\{a,b,c\}}(x) = ax^2 + bx + c
\]

\( a, b, c \), are determined by minimizing \( E \) (training = fitting by data)

\[
E = \frac{1}{2} \sum_d \left| f_{\{a,b,c\}}(x^{(d)}) - y^{(d)} \right|^2
\]

In physics language, variational method (with fitting)

Data determines a function form
Equivariance and convolution

Neural network works quite well in natural science


Score: Higher is better

Neural network wave function for many body (Carleo Troyer, Science 355, 602 (2017))

Neural net + "Expert knowledge" → Best performance
Equivariance and convolution

Knowledge $\ni$ Convolution layer = trainable filter, Equivariant

Filter on image

Convolution layer

Translational operation is *commutable* with filtering (equivariant)

Translational operation is *commutable* with convolutional neurons (equivariant)

This can be any filter which helps feature extraction (minimizing loss)

Equivariance reduces data demands. Ensuring symmetry (plausible Inference)
Equivariance and convolution

Convolutional Neural network have been good job but local

Convolutional neural layers in neural networks keep translational symmetry, it can be generalized to any continuous/discrete symmetry in the theory. It helps generalization.

However, 1 step of convolutional layer can pick up only local correlation and representability of neural networks is limited. Global correlations are sometimes important.

How can we overcome these difficulties?

Distant correlations here can be captured by 3 steps of convolutional operation (Repetition of local operation)
Transformer and Attention
Attention layer (in transformer model) has been introduced in a paper titled “Attention is all you need” (1706.03762)
State of the art architecture of language processing.
Attention layer is essential.
Attention layer can capture non-local correlations

Modifier in language can be non-local

Eg. I am Akio Tomiya living in Japan, who studies machine learning and physics

In physics terminology, this is non local correlation.

The attention layer enables us to treat non-local correlation with a neural net!

Schematic picture (in physics terminology)

\[ S_A = \sigma_{sm}(M)W^V S \sim \text{Weighted eff. ops.} \]

\[ M = KQ^\top \quad \text{Calculation of Attention score} \]

\[ Q = W^QS \sim \text{BST} \]
\[ K = W^KS \sim \text{BST} \]
\[ V = W^VS \sim \text{BST} \]

Queries \quad Key \quad Value
Transformer and Attention
Transformer shows scaling lows (power law)

Figure 1 Language modeling performance improves smoothly as we increase the model size, dataset size, and amount of compute used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

- Transformers requires huge data (e.g. GPT uses all electric books in the world)
  Because it has few inductive bias (no equivariance)
- It can be improved systematically (obey scaling law)
### Transformer and Attention

**Physically symmetric Attention layer**

Attention layer can capture global correlation

Equivariance reduces data demands for training

<table>
<thead>
<tr>
<th></th>
<th>Equivariance</th>
<th>Capturable correlation</th>
<th>Data demands</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convolution (∈ equivariant layers)</td>
<td>Yes 👍</td>
<td>Local 😞</td>
<td>Low 👍</td>
<td>Image recognition VAE, GAN Normalizing flow</td>
</tr>
<tr>
<td>(This work) Physically Equivariant attention</td>
<td>Yes 👍</td>
<td>Global 👍</td>
<td>?</td>
<td>This work arXiv: 2306.11527</td>
</tr>
</tbody>
</table>
Target system and its symmetry
Target system and its symmetry
Monte-Carlo + self-learning

Target system: Classical Heisenberg spin $S_i$ + Fermion on 2d lattice

$$H = -t \sum_{\alpha, \langle i, j \rangle} (\hat{c}_{i\alpha}^{\dagger} \hat{c}_{j\alpha} + \text{h.c.}) + \frac{J}{2} \sum_i S_i \cdot \hat{\sigma}_i$$

Symmetries:
- Global O(3)
- Translational
- 90-deg rotation

In lattice QCD language, Yukawa-theory with O(3) scalar field

$S_i : 3$ component scalar field on site $i$

$\hat{c}_{i\alpha} :$ Fermion (annihilation op.) at site $i$ with spin $\alpha$

Toy model for QCD.
Target system and its symmetry

Previous work

Target system: Classical Heisenberg spin $S_i$ + Fermion on 2d lattice

$$H = -t \sum_{\alpha, \langle i, j \rangle} (\hat{c}_i^\dagger \hat{c}_j + \text{h.c.}) + \frac{J}{2} \sum_i S_i \cdot \hat{\sigma}_i$$

Integrate out fermions $\hat{c}_{i\alpha}^\dagger, \hat{c}_{i\alpha}$:

$$Z = \sum_{\{S\}} \prod_n (1 + e^{-\beta(\mu - E_n(\{S\}))})$$

Non-local. Difficult. Time consuming

Using Hopping parameter expansion, we get a local effective model

This is used in a previous work:

$$H_{\text{Linear eff}}^\text{Linear} = - \sum_{\langle i, j \rangle} J_{\text{eff}} S_i \cdot S_j + E_0$$

Fit
Equivariant attention
Attention block makes effective spin field with non-local BST

\[ S_{\sim} = \]

\[ S = \]

Self-Attention block

Add & Norm

\[ S_A = \text{ReLU}(M)W^V S \]

\[ M = W^Q S (W^K S)^\top \]

\[ W^Q S \quad W^K S \quad W^V S \]

Self-Attention block

Smearing Rot. equivariant Trsl. equivariant trainable!
Self-learning Monte-Carlo

Attention block makes effective spin field with non-local BST

\[
S^{(l)} = \mathcal{N} \left( S^{(l-1)} + S_A \right) \quad \text{position-wise}
\]

\[
\mathcal{N}(S_i) = S_i / \|S_i\|
\]

\[
S_A = \text{ReLU}(M)W^V S
\]

\[
M = W^Q S (W^K S)^T
\]

\[
S_i \equiv \mathcal{N} \left( S^{(l-1)} + S_A \right)
\]

\[
\mathcal{N}(S_i) = S_i / \|S_i\|
\]

Self-Attention block

Add & Norm

Smeared fields
Rot. equivariant
Trsl. equivariant
Skip connection
Normalized!

Smearing
Rot. equivariant
Trsl. equivariant
trainable!
Self-learning Monte-Carlo
Variational Hamiltonian with Equivariant Attention layers

\[
S' \rightarrow H_{\text{eff}}
\]

\[
S(l) \equiv \mathcal{N} \left( S^{(l-1)} + S_A \right)
\]

\[\mathcal{N}(S_i) = S_i / \|S_i\|\]

\[
S_A = \text{ReLU}(M)W^V S
\]

\[
M = W^Q S(W^K S)^\top
\]

Smeared fields
Rot. equivariant
Trsl. equivariant
Skip connection
Normalized!

Smearing
Rot. equivariant
Trsl. equivariant
trainable!

Add & Norm
Self-Attention block
Add & Norm
Self-Attention block
Add & Norm
Self-Attention block

\[
S = \begin{array}{ccc}
\text{\textbullet} & \text{\textbullet} & \text{\textbullet} \\
\text{\textbullet} & \text{\textbullet} & \text{\textbullet} \\
\text{\textbullet} & \text{\textbullet} & \text{\textbullet} \\
\end{array}
\]
SLMC
= Self-learning Monte Carlo
= MCMC with variational Hamiltonian
Self-learning Monte-Carlo
SLMC = MCMC with an effective model

For statistical spin system, we want to calculate expectation value with
\[ W(\{S\}) \propto \exp[-\beta H(\{S\})] \]

On the other hand, an effective model \( H_{\text{eff}}(\{S\}) \) can compose MCMC,
\[ \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \quad \text{this distributes} \quad W_{\text{eff}}(\{S\}) \propto \exp[-\beta H_{\text{eff}}(\{S\})] \]
if the update 「→」 satisfies the detailed balance. We can employ Metropolis test like
\[ A_{\text{eff}}(\{S\}', \{S\}) = \min\left(1, \frac{W_{\text{eff}}(\{S\}')}{W_{\text{eff}}(\{S\})}\right). \]
Self-learning Monte-Carlo

SLMC = MCMC with an effective model

For statistical spin system, we want to calculate expectation value with

\[ W(\{S\}) \propto \exp[-\beta H(\{S\})] \]

On the other hand, an effective model \( H_{\text{eff}}(\{S\}) \) can compose MCMC,

\[ \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \]

this distributes \( W_{\text{eff}}(\{S\}) \propto \exp[-\beta H_{\text{eff}}(\{S\})] \)

if the update \( \rightarrow \) satisfies the detailed balance. We can employ Metropolis test like

\[ A_{\text{eff}}(\{S'\}, \{S\}) = \min \left( 1, \frac{W_{\text{eff}}(\{S'\})}{W_{\text{eff}}(\{S\})} \right). \]

**SLMC:** Self-learning Monte-Carlo

We can construct *double* MCMC with \( H(\{S\}) \) and \( H_{\text{eff}}(\{S\}) \)

\[ \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \rightarrow \{S\} \]

with Metropolis-Hastings test: \( A(\{S'\}, \{S\}) = \min \left( 1, \frac{W(\{S'\})}{W(\{S\})} \right) \frac{W_{\text{eff}}(\{S\})}{W_{\text{eff}}(\{S'\})} \).

- Effective model can have fit parameters
- **Exact!** It satisfies detailed balance with \( W(\{S\}) \)
- It has been used for full QCD too (arXiv: 2010.11900, 2103.11965)
Self-learning Monte-Carlo
Monte-Carlo + self-learning

Target system: Classical Heisenberg spin $S_i$ + Fermion on 2d lattice

$$H = - t \sum_{\alpha, \langle i, j \rangle} (\hat{c}_{i\alpha}^{\dagger} \hat{c}_{j\alpha} + \text{h.c.}) + \frac{J}{2} \sum_i S_i \cdot \hat{\sigma}_i$$

Symmetries:
- Global O(3)
- Translational
- 90-deg rotation

$$[\hat{\sigma}_i]_\gamma \equiv \hat{c}_{i\alpha}^{\dagger} \sigma^\gamma_{\alpha\beta} \hat{c}_{i\beta}$$

$$S_i = \begin{pmatrix} S_i^1 \\ S_i^2 \\ S_i^3 \end{pmatrix}^T$$

$$S_i^\mu \in \mathbb{R}$$

In lattice QCD language, **Yukawa-theory with O(3) scalar field**

- $S_i$: 3 component scalar field on site $i$
- $\hat{c}_{i\alpha}$: Fermion (annihilation op.) at site $i$ with spin $\alpha$

Toy model for QCD.
Self-learning Monte-Carlo

Previous work

Target system: Classical Heisenberg spin $S_i$ + Fermion on 2d lattice

$$H = -t \sum_{\alpha,\langle i,j \rangle} (\hat{c}_{i\alpha}^\dagger \hat{c}_{j\alpha} + \text{h.c.}) + \frac{J}{2} \sum_i S_i \cdot \hat{\sigma}_i$$

Integrate out fermions $\hat{c}_{i\alpha}^\dagger, \hat{c}_{i\alpha}$:

$$Z = \sum_{\{S\}} \prod_n (1 + e^{-\beta(\mu - E_n(\{S\}))})$$

Non-local. Difficult. Time consuming

(In previous work), using Hopping parameter expansion, we can get a local effective model:

$$H_{\text{eff}}^{\text{Linear}} = - \sum_{\langle i,j \rangle} J_{\text{eff}} S_i \cdot S_j + E_0$$

Fit

In SLMC,
Poor scaling, poor representability = poor acceptance!
Self-learning Monte-Carlo

Equivariant Attention layer

We can construct effective hamiltonian with output of Attention layer because “output of Attention = smeared fields with non-local correlation”

$$H_{\text{Linear}}^{\text{Linear}} = - \sum_{(i,j)} J_{\text{eff}}^{\text{eff}} S_i^{\text{eff}} \cdot S_j^{\text{eff}} + E_0$$

$$S^{(l)} \equiv \mathcal{N} \left( S^{(l-1)} + S_A \right)$$

Smeared fields
Rot. equivariant
Trsl. equivariant
trainable!

Smeared fields
Rot. equivariant
Trsl. equivariant
trainable!
Results
Transformer and Attention

Acceptance rate is improved with # of layers

Acceptance rate

<table>
<thead>
<tr>
<th>Num. of attention layers</th>
<th>Acceptance ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0.2</td>
</tr>
<tr>
<td>2</td>
<td>0.4</td>
</tr>
<tr>
<td>3</td>
<td>0.6</td>
</tr>
<tr>
<td>4</td>
<td>0.8</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Models with the attention

(transform as prev. work)

Transformers

Linear

Num. of attention layers

= # of parameters

Note: As far as we tested,
CNN-type does not work in this case.
No improvements with increase of layers.
(Global correlations of fermions from
Fermi-Dirac statistics make acceptance bad?)

Obsevables

Physical values are consistent
(as we expected)

Nx=Ny=6
(Lattice sites)
Transformer and Attention

Acceptance rate -> MSE (~ loss), Scaling law (power law)

acceptance = \exp\left(-\sqrt{\text{MSE}}\right)

Estimated MSE

Model w/o attention

Models with the attention

fit \sim (7.1/x)^{1.1}

num. of trainable parameters

(1 layer \sim 30 parameters)

arXiv: 2306.11527 + update

Preliminary
Summary

Physics + Machine learning

- Equivariance helps generalization of machine learning models
  Attention enables us to capture global correlations

- O(3) spin-fermion system can be efficiently simulated SLMC with Attention
  - In lattice QCD terminology, it is O(3) scalar + fermions
  - Increase of # of attention layers makes increase acceptance rate
  - Models with the CNN-type do not work (not showed)
  - SLMC with the equivariant Attention shows the scaling law

- Attention is all you need (?)

Future work:

- Apply “equivariant attention” on full QCD
- What is “gauge equivariant attention”? Is it possible?
- Can we marge it with gauge covariant convolution? (arXiv: 2103.11965)
- Can we use this to the flow based sampling algorithm? (GomalizingFlow.jl)

Thanks!
Details
Self-learning Monte-Carlo

Attention layer

$$S' \rightarrow H_{\text{eff}} = \text{tr}[S'(JS')^T]$$

$$S^{(l)} \equiv \mathcal{N} \left( S^{(l-1)} + \text{SelfAttention}_{\theta^{(l)}}^{\text{spin}}(S^{(l-1)}) \right)$$

$$\mathcal{N}(S_i) = \frac{S_i}{\|S_i\|}$$
Self-learning Monte-Carlo

Equivariant under spin-rotation & translation

\[ S = \left( S_1^T \quad S_2^T \quad S_3^T \quad S_4^T \right)^T \]

\[ S_i : \text{Classical Heisenberg spin at site } i \]

\[ S : \text{A spin configuration} \]

**Gram matrix**

\[ G \equiv S^T S = \begin{pmatrix}
S_1^T S_1 & S_1^T S_2 & S_1^T S_3 & S_1^T S_4 \\
S_2^T S_1 & S_2^T S_2 & S_2^T S_3 & S_2^T S_4 \\
S_3^T S_1 & S_3^T S_2 & S_3^T S_3 & S_3^T S_4 \\
S_4^T S_1 & S_4^T S_2 & S_4^T S_3 & S_4^T S_4
\end{pmatrix} \]

- G is a matrix for coordinate but not for spin.
- Spin rotation for Si keeps G invariant.

**3 component scalar, normalized**

\[ S_i^T = \left( s_i^1 \quad s_i^2 \quad s_i^3 \right)^T \]

\[ |S_i| = \sqrt{(s_i^1)^2 + (s_i^2)^2 + (s_i^3)^2} = 1 \]

If an effective hamiltonian is a function Gram matrix, it has rotational symmetry
Self-learning Monte-Carlo
Equivariant under spin-rotation & translation

\[ S = \left( S_1^T, S_2^T, S_3^T, S_4^T \right)^T \]

- Local weighted sum over neighbors
= “Smeared spin” with parameters
~ “Block spin sum” with parameters

\[ \tilde{S}_i^\alpha = \sum_{l=0}^{\infty} W^\alpha_l S_{i+l} \quad \alpha = \text{Q, K, V} \]

\[ W^\alpha_l \in \mathbb{R} : \text{trainable} \]

Translationally equivariant
Rotationaly equivariant

\[ S_i^T = \left( s_i^1, s_i^2, s_i^3 \right)^T \]

\[ |S_i| = \sqrt{(s_i^1)^2 + (s_i^2)^2 + (s_i^3)^2} = 1 \]

3 component scalar, normalized
Self-learning Monte-Carlo
Equivariant under spin-rotation & translation

\[ S = \begin{pmatrix} S_1^T & S_2^T & S_3^T & S_4^T \end{pmatrix}^T \]

\[ S_i^T = (s_i^1, s_i^2, s_i^3)^T \]

\[ \tilde{S}_i^\alpha = \sum W_\alpha^\alpha S_{i+l} \] “Smeared spin”

Gram matrix with smeared spin

\[ M = \tilde{G}^\alpha \equiv (\tilde{S}^\alpha)^\top \tilde{S}^\alpha \quad \alpha = Q, K, V \]

Translationally covariant
Rotatinally invariant

\[ S_A = \text{ReLU}(M)W^VS \]

\[ = \text{ReLU}(M)\tilde{S}^V \]
Self-learning Monte-Carlo

Attention layer

\[ S' \rightarrow H_{\text{eff}} = \text{tr}[S'(JS')^T] \]

\[ S^{(l)} = N \left( S^{(l-1)} + \text{SelfAttention}_{\theta^{(l)}}(S^{(l-1)}) \right) \]

\[ N(S_i) = \frac{S_i}{\|S\|} \]

Note that, if \( W = 0 \), it fall back to identity operation.

Effective Hamiltonian fall back to a linear model.
Transformer and Attention
Training scheme

\[ S' \rightarrow H_{\text{eff}} = \text{tr}[S'(JS')^T] \]

\[ S = \begin{pmatrix} \vdots \\ \vdots \end{pmatrix} \]

\[ S' = \begin{pmatrix} \vdots \\ \vdots \end{pmatrix} \]

\[ S = \begin{pmatrix} \vdots \\ \vdots \end{pmatrix} \]