Lorentz-Equivariant Geometric Algebra Transformers (LGATr) for FCCee Physics

entsprechend den Gestaltungsrichtlinien von Februar 2025

Ánuar Sifuentes Name | 11. December 2025

ETP – Institut für experimentelle Teilchenphysik

Content

1. **Problem & Motivation**
2. **How LGATr Works?**
3. **Why does this matter for FCCee and for $B_s \rightarrow \tau^+\tau^-$?**

Problem & Motivation

Challenges in Particle Physics ML

HEP imposes strict requirements on ML models

- Must learn **complex physical relations** with high precision
- Need **data efficiency** and **fast inference/training**
- Standard ML models **ignore the structure and symmetries** of HEP data
- Particle interactions follow **QFT** and must respect fundamental symmetries
- Especially **Lorentz symmetry** from special relativity

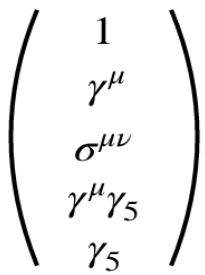
Image source: https://iticollege.edu/blog/machine-learning-basics-and-real-world-applications/

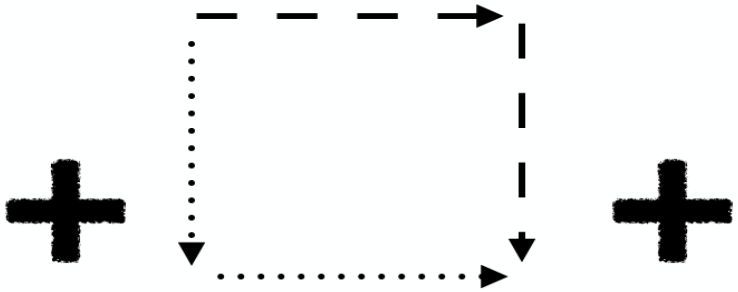
Limitations of current tools

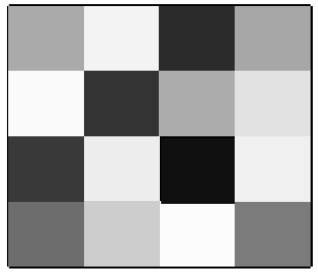
Tool	Handles variable inputs?	Respects Lorentz Symmmetry?	Data Efficiency
BDT	❌ No	❌ No	Low
MLP	❌ No	❌ No	Medium
CNN	✅ Yes (as images)	❌ No	Medium
Transformer	✅ Yes	❌ No	High
LGATr	✅ Yes	✅ Yes	High

How does Lorentz-Equivariant Geometric Algebra Transformers (LGATr) work?

General Overview



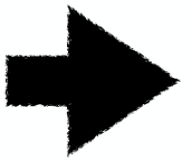


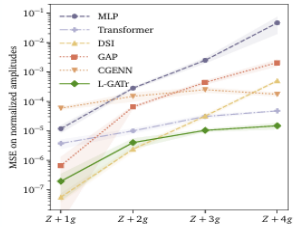


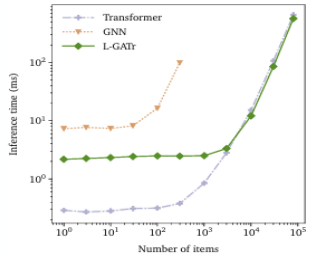
Geometric algebra representations

Equivariant layers

Transformer architecture







GATr was originally developed for E(3) arXiv:2305.18415

Strong performance on diverse problems

Scalable to thousands of tokens

3

Image source: Chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://indico.kit.edu/event/4479/contributions/17664/attachments/7962/12858/lgatr-crc.pdf

Core Idea 1 – Geometric Algebra (GA)

- **Also called Clifford algebra**: extends a vector space (e.g., $\mathbb{R}^3$or Minkowski space) with **multivectors** (scalars, vectors, bivectors, etc.).
- Built on the **geometric product** = inner product + generalized cross/outer product.
- Spacetime GA $G_{1,3}$is **16-dimensional**:
  -grade 0: scalars (type info),
  -grade 1: four-vectors (p^{μ},(
  -higher grades: antisymmetric tensors → richer representations.

Why L-GATr uses GA

- GA enables **equivariance**: GATr → E(3), L-GATr → **Lorentz** transformations.
- Each particle = a **GA token** combining type (scalar channel) + four-momentum (grade-1).
- Lets the network operate directly on **geometric objects**, not arbitrary feature vectors.

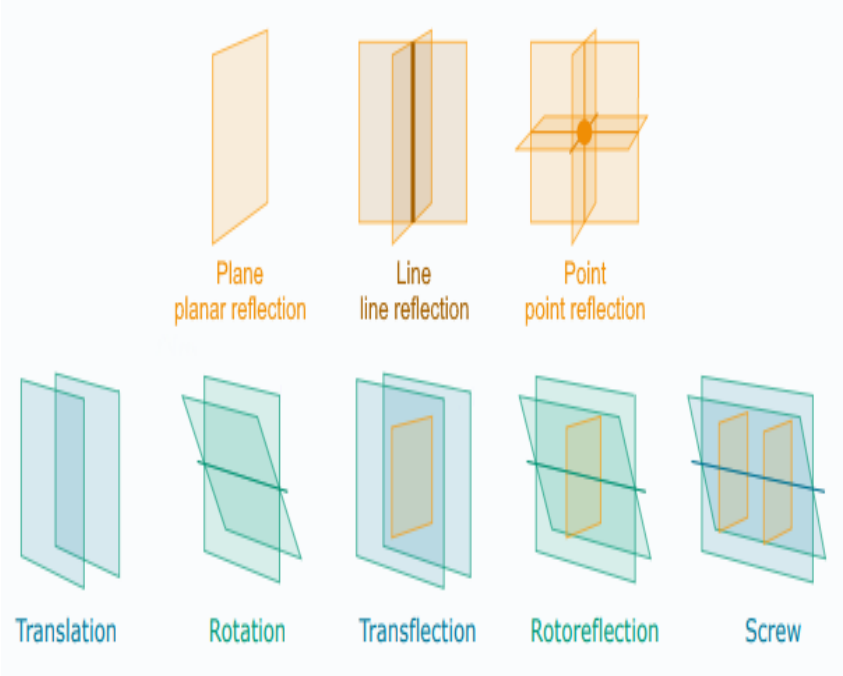


Image source: https://en.wikipedia.org/wiki/Plane-based_geometric_algebra

Core Idea 2 – Lorentz equivariance in Action

$$\Lambda \in \mathrm{SO}^+(1,3):$$
$$f(\Lambda \cdot x) = \Lambda \cdot f(x)$$

Lorentz Equivariance

- Multivector tokens (grades 0–4) → each grade transforms consistently under Lorentz transformations
- Linear layers, attention, LayerNorm, and nonlinearities are **adapted for multivectors** while preserving equivariance
  - -Per-grade linear mixing (optional γ_5 term)
  - -Minkowski inner-product attention
  - -Scalar-gated GELU + geometric product

Partial Symmetry Breaking

- LHC beam & detector break full Lorentz symmetry → fully equivariant networks may miss physical asymmetries.
- **Reference multivectors** allow controlled symmetry breaking:
  - -Reduce symmetry to SO(3), SO(2), etc. depending on beam/detector geometry
  - -Treated as architectural constants, not input data
  - -Network can selectively use or ignore them

Core Idea 3: Transformers

Overall structure:
•Stacked **Blocks**: Block = Attention + MLP
•Full network: L-GATr(x) = Linear → Block → … → Block → Linear

AttentionBlock:
•Captures **relations between all particle tokens**
•Linear projections → **Minkowski inner-product attention** → output + residual
•Weighs particles **context-dependently**
•**Lorentz-equivariant**

•**MLPBlock:** Linear → Gated GELU → Linear → Geometric Product → + residual

•**Key points:**
•Tokens = **multivectors + scalars**
•Residuals + LayerNorm stabilize training
•Geometric product + multivector layers → expressivity for 4-momenta

Purpose:
•Capture **complex particle interactions** beyond standard NNs

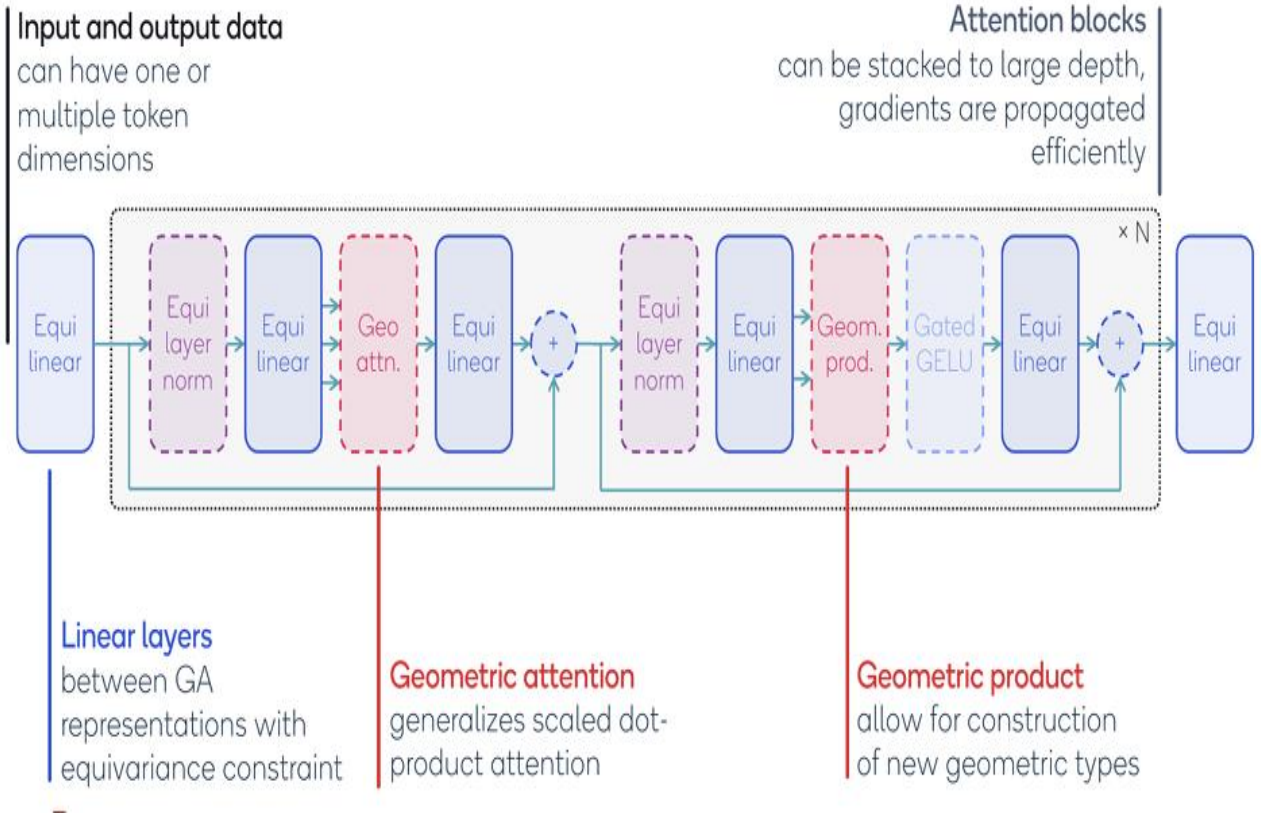
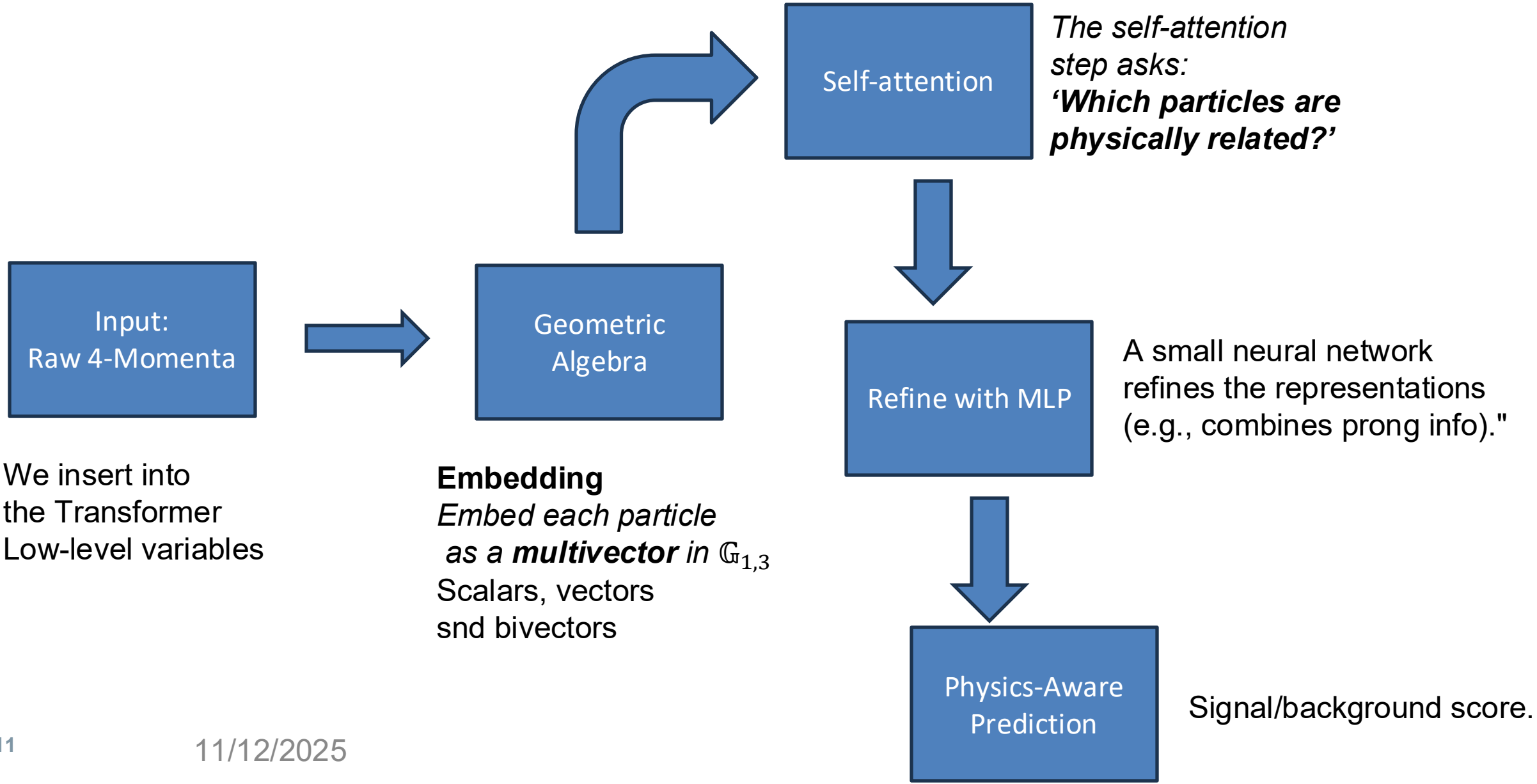


Image source: Chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://indico.kit.edu/event/479/contributions/17664/attachments/7962/12858/lgatr-crc.pdf

Putting all together (simplified)



The self-attention step asks: ***'Which particles are physically related?'***

A small neural network refines the representations (e.g., combines prong info)."

We insert into the Transformer Low-level variables

Embedding
Embed each particle as a ***multivector*** *in* $\mathbb{G}_{1,3}$
Scalars, vectors snd bivectors

Signal/background score.

Why does this matter for FCCee and for $B_s \rightarrow \tau^+\tau^-$?

Advantages for FCCee & $B_s \rightarrow \tau^+\tau^-$

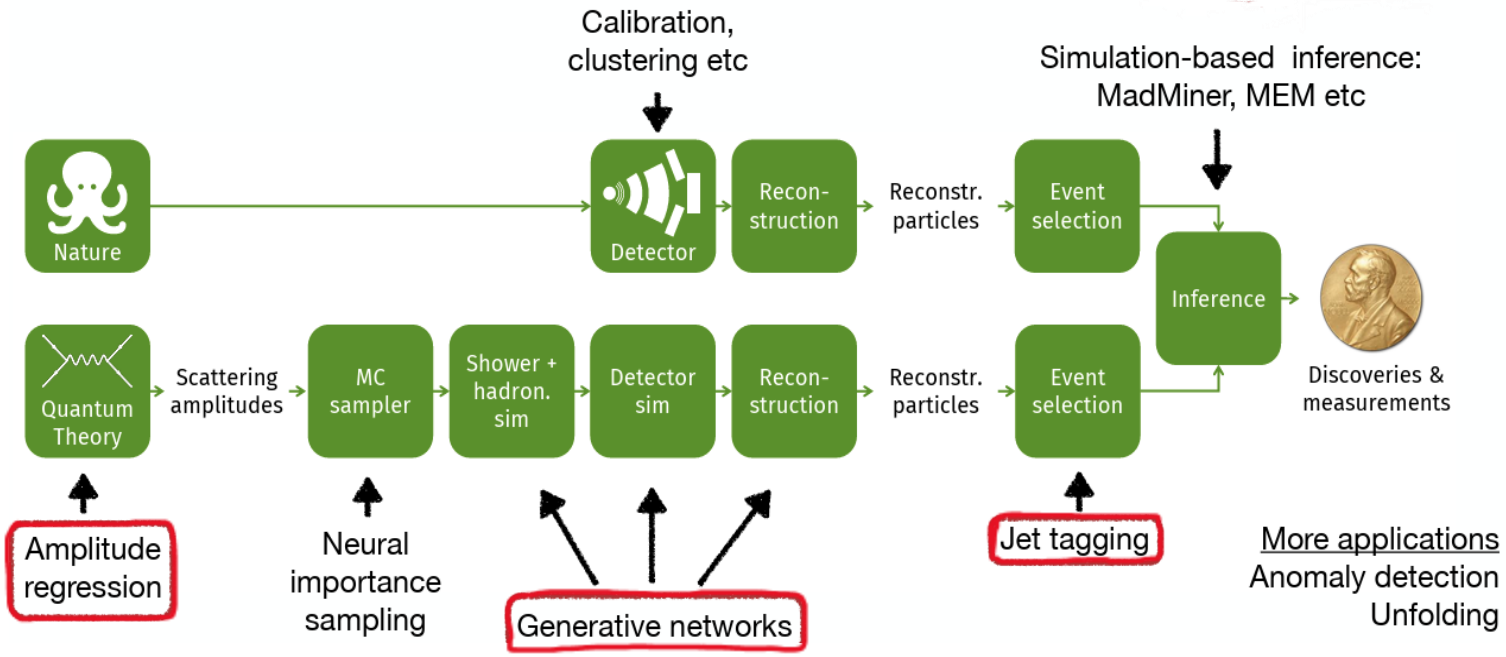


Image source: Chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://indico.kit.edu/event/4479/contributions/17664/attachments/7962/12858/lgatr-crc.pdf

Strong Performance on:
- Amplitude regression
- Generative Networks
- Jet Tagging

Data efficiency: Learns from fewer $B_s \rightarrow \tau\tau$ events.

We can use full 4-momentum structure

The Transformer's attention lets LGATr to look for patterns
- Essential for rare-decay discrimination

No need to define high-level variables

Summary + Practical Next Steps

Summary

- **LGATr** is a relatively new (*23 May 2024*) and promising ML architechture for HEP
- It is **Lorentz equivariant** and it has the possibility to break symmetries if necessary
- The architechture uses **Geometric Algebra (GA),** greatly improving expressivity
- Because **GA** embeds the symmetries in its multiplication rules, it becomes Lorentz-equivariant
- This ML method is a **Transformer**, taking advantage of the **Attention Mechanism**
- **LGATr has strong performance** on Amplitude regression, Generative Networks and Jet Tagging

Practical Next Steps

- **Understand the code** behind LGATr
- Of course test this ML Architecture on $\mathbf{B_s} \rightarrow \tau^+\tau^-$
- Try different combinations of **low-level variables**
- **Outperform Marc's Groeninger** benchmark

Thank you for your attention!

Extra 1: Amplitude Regression

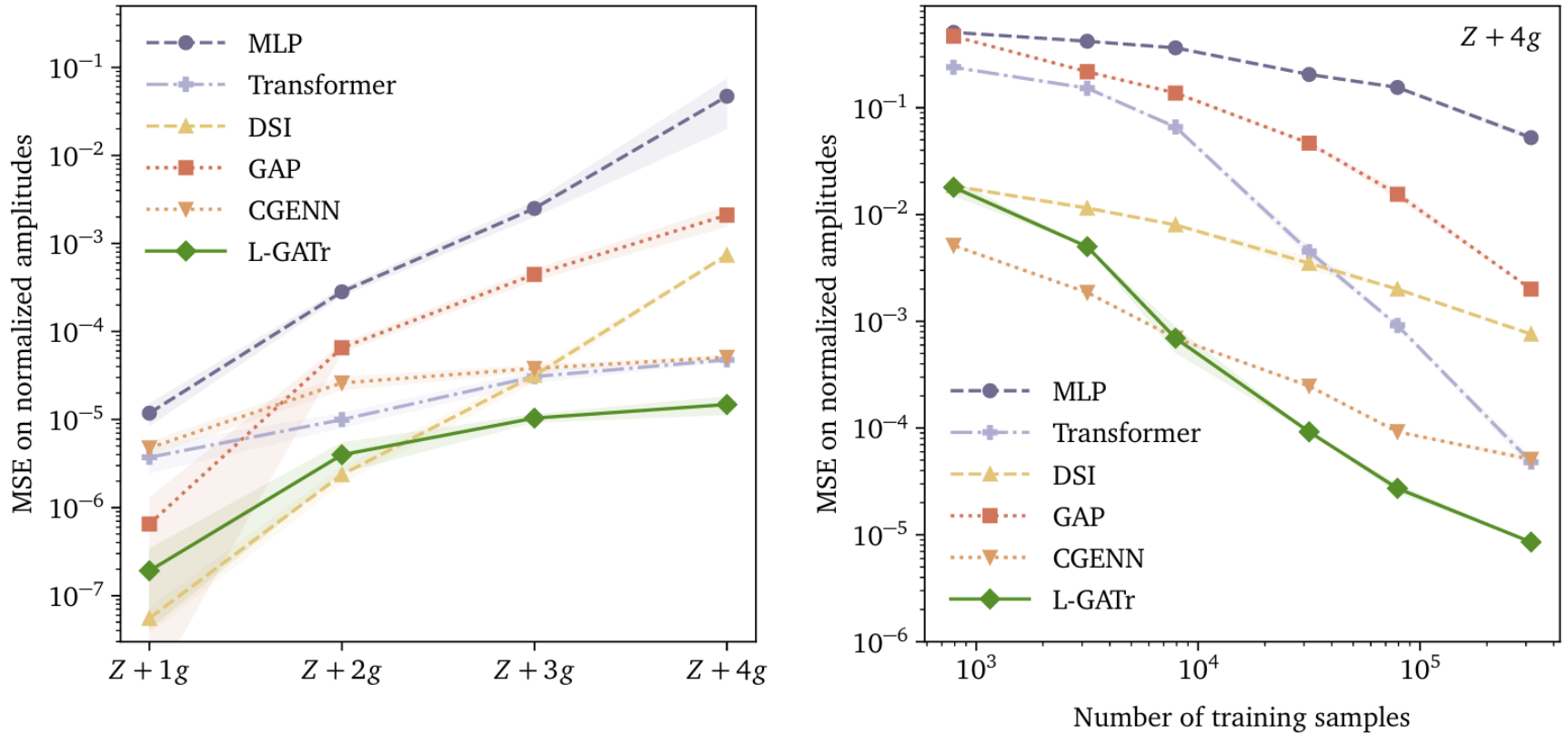


Figure 3: Amplitude surrogates. **Left**: Surrogate error for processes of increasing particle multiplicity and complexity, training on the full dataset of $4 \cdot 10^5$ samples. L-GATr outperforms the baselines, especially at more complex processes. **Right**: Surrogate error as a function of the training dataset size.

Image source: Lorentz-Equivariant Geometric Algebra Transformers for High-Energy Physics.
https://arxiv.org/abs/2405.14806

Extra 2: Generative Networks

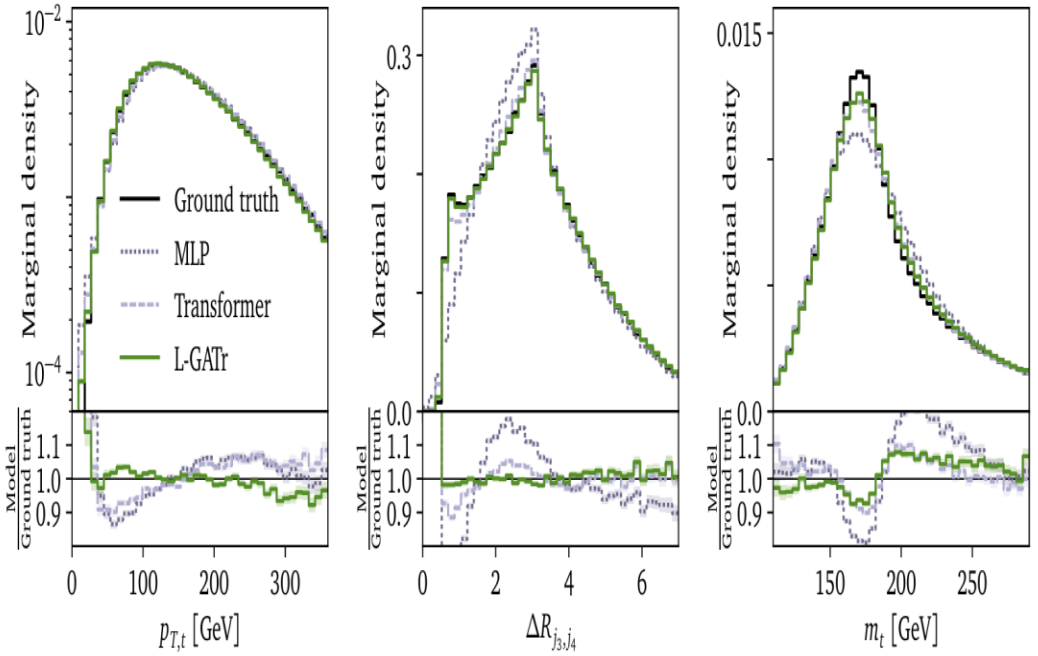


Figure 4: Generative modelling: Marginal distributions of reconstructed particles in the $pp \to t\bar{t} + 4$ jets process. We compare the ground-truth distribution (black) to three generative models: continuous normalizing flows based on a Transformer, MLP, or our L-GATr network. The three marginals shown represent kinematic features that are known to be challenging. The L-GATr flow describes them most accurately.

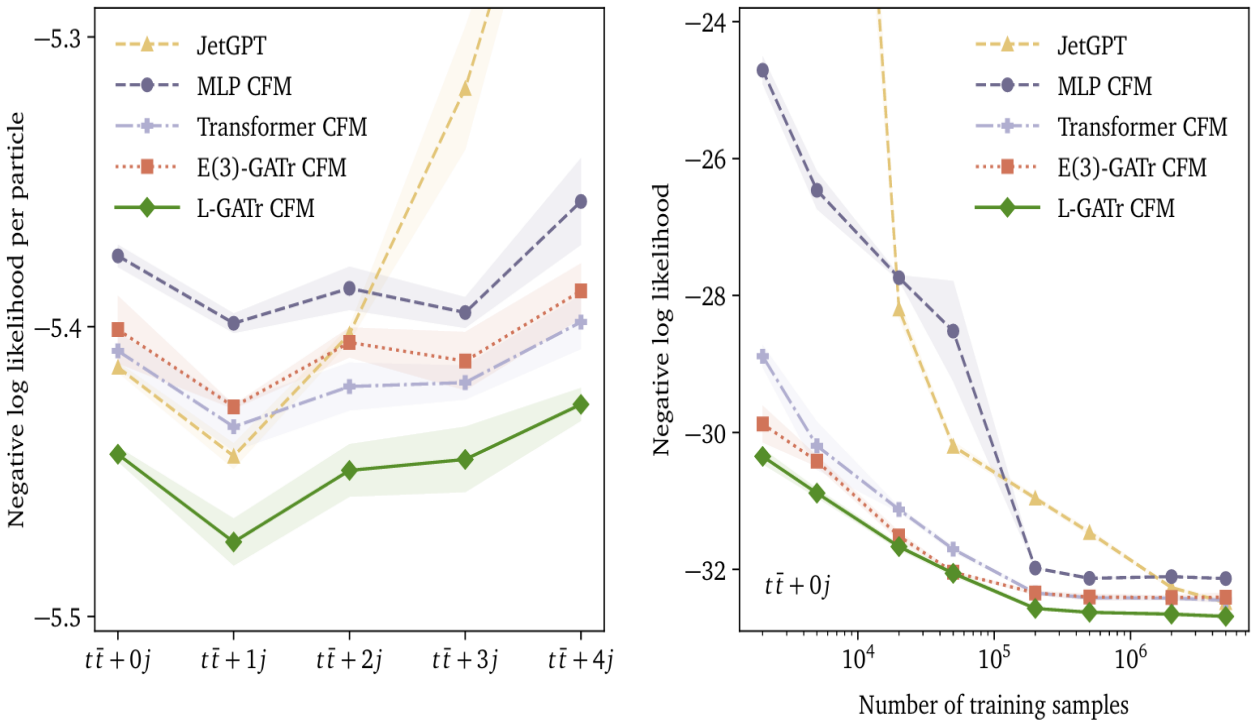


Figure 5: Generative modelling: negative log likelihood on the test set (lower is better). **Left**: For different processes. **Right**: As a function of the training dataset size. We show the mean and standard deviation of three random seeds. The L-GATr conditional flow matching (CFM) model outperforms all other CFM models as well as the autoregressive transformer JetGPT, across all processes and all training set sizes.

Image source: **Lorentz-Equivariant Geometric Algebra Transformers for High-Energy Physics.**
https://arxiv.org/abs/2405.14806

Extra 3: Jet Tagging

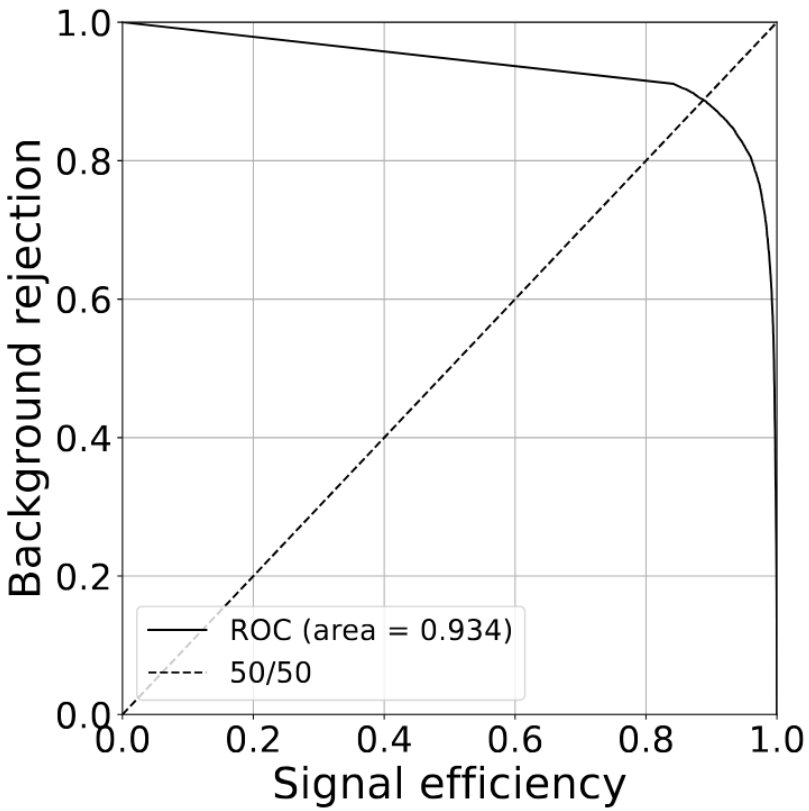
Model	Accuracy	AUC	$1/\epsilon_B$ ($\epsilon_S = 0.5$)	$1/\epsilon_B$ ($\epsilon_S = 0.3$)
TopoDNN [49]	0.916	0.972	–	295 ± 5
LoLa [16]	0.929	0.980	–	722 ± 17
P-CNN [1]	0.930	0.9803	201 ± 4	759 ± 24
N-subjettiness [62]	0.929	0.981	–	867 ± 15
PFN [51]	0.932	0.9819	247 ± 3	888 ± 17
TreeNiN [58]	0.933	0.982	–	1025 ± 11
ParticleNet [64]	0.940	0.9858	397 ± 7	1615 ± 93
ParT [65]	0.940	0.9858	413 ± 16	1602 ± 81
LorentzNet* [42]	0.942	0.9868	498 ± 18	2195 ± 173
CGENN* [68]	0.942	0.9869	500	2172
PELICAN* [10]	**0.9426** ± 0.0002	**0.9870** ± 0.0001	–	**2250** ± 75
L-GATr (ours)*	0.9423 ± 0.0002	**0.9870** ± 0.0001	**540** ± 20	2240 ± 70

Table 1: Top tagging. We compare accuracy, area under the ROC curve (AUC), and inverse background acceptance rate $1/\epsilon_B$ at two different signal acceptance rates (or recall) $\epsilon_S \in (0.3, 0.5)$ for the top tagging dataset from Kasieczka et al. [50]. Lorentz-equivariant methods are indicated with an asterisk*; the best results for each metric are in **bold**. For L-GATr, we show the mean and standard deviation of five random seeds. Baseline results are taken from the literature.

Image source: **Lorentz-Equivariant Geometric Algebra Transformers for High-Energy Physics.**
https://arxiv.org/abs/2405.14806

Extra 4: Marc Groeninger's work

All Available Features



Only pion information

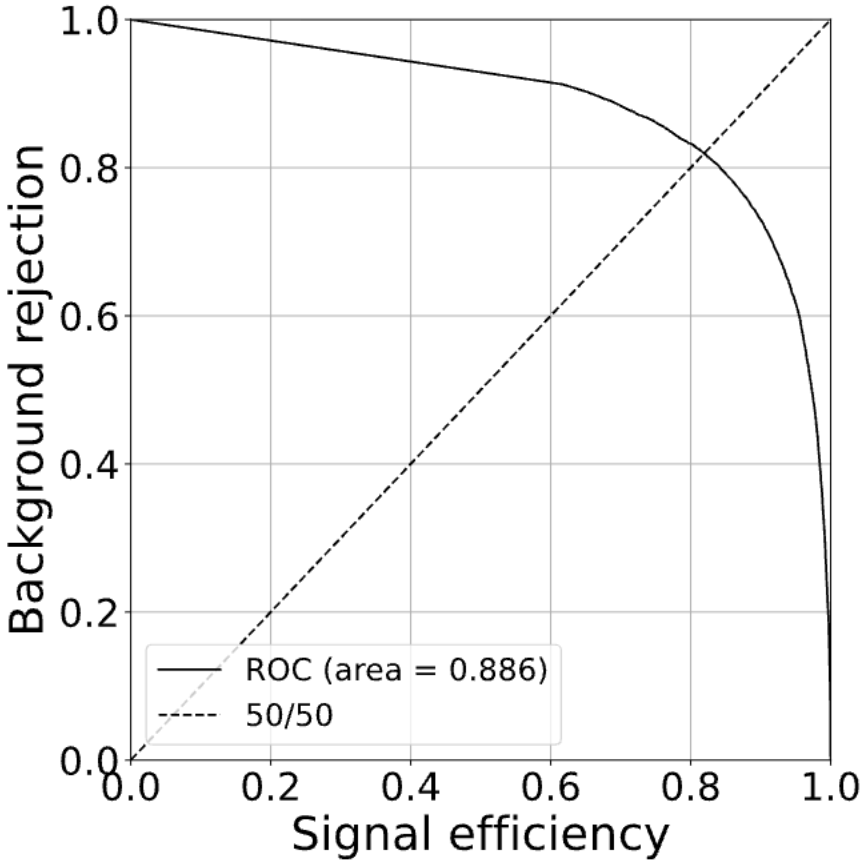


Figure 4.2.: ROC curve of the first L-GATr model.

Image source: Marc Groeninger's Bachelor Thesis. https://www.overleaf.com/project/681bc1aa75868d894e4a0c78