Neural Scaling Laws for Jet Tagging with Particle Transformers

Sitian Qian
Department of Physics
University of Wisconsin-Madison
Madison, WI 53706
sitian.qian@cern.ch

Cheng Jiang
School of Physics and Astronomy
University of Edinburgh
Edinburgh EH9 3FD, United Kingdom
chjiang@cern.ch

Alexandre De Moor
Inter-university Institute for High Energies
Vrije Universiteit Brussel
1050 Brussels, Belgium
alexandre.thierry.de.moor@cern.ch

Pavlo Kashko
Inter-university Institute for High Energies
Vrije Universiteit Brussel
1050 Brussels, Belgium
pavlo.kashko@cern.ch

Ya-Feng Lo
Fundamental Physics Directorate
SLAC National Accelerator Laboratory
Menlo Park, CA 94025, USA
ylo@cern.ch

Abstract

While neural scaling laws are well-established in natural language and vision, their behavior in domains governed by physical constraints remains largely unexplored. We present a systematic, Chinchilla-style scaling study of the Particle Transformer (ParT) for jet tagging using the JetClass dataset. By evaluating across a broad grid of model sizes and training budgets, we show that ParT exhibits a notably steeper data scaling curve than prior baselines, reflecting efficient usage of training data. Furthermore, we isolate sample scaling through a strict single-epoch (no-reuse) regime, which reaches a higher performance ceiling than multi-epoch training, and we find the fitted exponents to be consistent across Pythia, Herwig, and their mixture within statistical uncertainty. Finally, we introduce a mathematical framework that propagates the cross-entropy fit to downstream physical metrics, including accuracy, AUC, and background rejection, helping guide how to allocate compute and data for future foundation models in high energy physics.

1 Introduction

The scale and complexity of data at the Large Hadron Collider (LHC) [1] make it a natural setting for modern machine learning in high energy physics [2]. Jet tagging is one of the most important tools for new physics searches, identifying the fundamental particles that produce collimated sprays of hadrons in the detector [3]. Over the past decade, jet tagging has shifted from relying on hand-crafted high-level observables to learning directly from low-level particle kinematics. Graph Neural Networks (GNNs), such as ParticleNet [4], and more recently architectures like the Particle Transformer (ParT) [5], have steadily improved performance by treating jets as point clouds or fully connected sets of particles.

Following the success of large language models, there is growing interest in developing foundation models for physics [6]. The idea is to pretrain high-capacity models on large simulated datasets

40th Conference on Neural Information Processing Systems (NeurIPS 2026). Workshop: PAI2026.

such as the JetClass dataset [7] to learn general representations that can be adapted to downstream tasks. However, training such models is computationally expensive. To better understand how to allocate resources, the machine learning community studies neural scaling laws [8], which describe how performance depends on model size N and dataset size D. Hoffmann et al. [9] proposed the Chinchilla scaling form:

$$L(N, D) = AN^{-\alpha} + BD^{-\beta} + E, \tag{1}$$

where α and β are the scaling exponents for model size and data, and E represents the irreducible error [10].

While scaling laws are well established in language and vision [11–14], their behavior in physics remains less understood. Vigl et al. [15] showed that a standard Set Transformer trained on JetClass follows the Chinchilla form. A key open question is whether a more specialized architecture behaves differently. The ParT incorporates physically motivated pairwise features, such as kinematic distances, directly into the attention mechanism. It is not yet clear whether this inductive bias changes the scaling exponents α and β.

Physics data is also generated by Monte Carlo simulation, which introduces model-dependent assumptions. We study ParT across a dense grid of (N, D) and examine how the generator affects the irreducible error E. The public JetClass dataset [7] uses Pythia [16]; we additionally simulate a matched Herwig [17] sample (same processes, selections, and feature definitions) and a "Mixed" set combining the two equally. Each generator provides 100M unique jets, so larger budgets reuse data; to isolate the true data exponent β we complement multi-epoch training with a strict single-epoch ("1-ep") setting in which all samples are unique.

2 Related Work

Deep Learning for Jet Tagging. Jet tagging [18–30] has benefited significantly from low-level particle kinematics. Early approaches using convolutional networks on jet images [31] and recurrent models on particle sequences [32] have largely given way to permutation-invariant architectures that treat jets as point clouds. Graph Neural Networks, in particular ParticleNet [4], showed the importance of learning local geometric structure. More recently, Transformer models have achieved state-of-the-art performance. The ParT [5] extends self-attention by incorporating pairwise particle features, such as kinematic distances, directly into the attention mechanism.

Scaling Laws in Physics. As high energy physics moves toward large-scale foundation models [6, 33–41], understanding data and compute requirements becomes essential. Yet scaling studies in this setting remain limited. Vigl et al. [15] analyzed a Set Transformer on JetClass and found Chinchilla exponents of $\alpha \approx 0.44$, $\beta \approx 0.22$, with an error floor $E \approx 0.32$. We extend this line of work by studying a more expressive architecture, ParT, and examining whether its inductive bias changes data scaling behavior. We also connect scaling laws to experimentally relevant quantities by mapping cross-entropy fits to metrics such as AUC and background rejection.

3 Method

Fitting procedure. We fit Eq. 1 using `scipy.optimize.curve_fit`, enforcing non-negativity of (A, B, E) and constraining $\alpha, \beta \leq 2$. We report the coefficient of determination R^2 evaluated on the observed grid points. N and D are expressed in millions of parameters and training samples seen (i.e., summed over epochs in the multi-epoch setting). To assess the robustness of the fitted exponents, we propagate the covariance matrix returned by the fit to obtain 1σ errors on (A, α, B, β, E), and we provide a complementary Bayesian fit that places weakly-informative priors on all parameters and samples the posterior with NUTS, reporting credible intervals on α and β that can be compared directly with values in the literature.

Models, inputs, and data budgets. Table 1 summarizes the seven ParT variants. All variants use the full ParT input set: per-particle features (kinematics and particle identification) together with the pairwise interaction features (energy/momentum-baseyd distances between particles) injected into the attention, exactly as in the original ParT [5]. Throughout, we label a configuration as `generator/feature-set/pairwise`; for example, `Pythia/full/pair` denotes Pythia samples, the full feature set, and pairwise features enabled. Training budgets are

Table 1: ParT variants (full features, pairwise interactions on). Columns: per-block embedding dimensions, pairwise-feature embedding dimensions, number of attention heads h, number of transformer blocks l, and total parameter count.

Name	embed dims	pair embed	h	l	params
nano	[32,128,32]	[16,16,16]	4	2	0.1M
micro	[48,192,48]	[24,24,24]	4	3	0.2M
tiny	[64,256,64]	[32,32,32]	4	4	0.4M
small	[96,384,96]	[48,48,48]	8	6	1.0M
base	[128,512,128]	[64,64,64]	8	8	2.1M
large	[256,1024,256]	[128,128,128]	16	12	8.5M
xlarge	[384,1536,384]	[192,192,192]	16	16	25M

Table 2: Loss→metric maps $\log(1-m) = aL+b$ (Eq. 2), fit separately per regime ($R^2 \gtrsim 0.98$ each). †Uses $\log R_{\text{bkg}}$.

	multi-ep		1-ep	
Metric	a	b	a	b
Val. acc.	1.78	−2.65	0.75	−2.09
Val. ROC AUC	3.09	−5.58	1.34	−4.64
Bkg. rej.†	−3.09	5.58	−1.34	4.64

$D \in \{5, 10, 25, 50, 100, 250, 500\}$M samples seen; in the 1-ep setting each unique jet is seen exactly once. The JetClass public release distributes Pythia files; the Herwig companion used for the headline fits is a matched dataset produced with identical event selections and feature definitions, made available through H. Qu's distribution chain.[1] A Sherpa companion is in preparation.

Bounded metrics and loss–metric maps. Besides the cross-entropy loss L, we track two metrics on the held-out validation split: *validation accuracy* (10-class top-1) and *validation ROC AUC* (a single scalar: the macro-averaged one-vs-one (OvO) multiclass ROC AUC over all $\binom{10}{2}$ class pairs); from the latter we derive the *background rejection* $R_{\text{bkg}} = 1/(1 - \text{ROC AUC})$. Cross-entropy loss is unbounded above and decreases as models improve, making it directly compatible with the Chinchilla form, whereas bounded metrics approach 1. We therefore apply the transformation $y \equiv \log(1 - m)$ for a metric $m \in [0, 1)$, so that $y \to -\infty$ as $m \to 1$, analogous to decreasing loss.

Empirically, after this transformation the metrics are approximately linear in the cross-entropy loss L:

$$\log(1 - m) = a\,L + b. \tag{2}$$

Because reusing data shifts the loss–metric relation, we fit Eq. 2 *separately* for the multi-epoch and single-epoch regimes (Table 2); each attains $R^2 \gtrsim 0.98$. Given a fitted loss $L(N, D)$, composing with the appropriate map yields scaling curves for accuracy, ROC AUC, and background rejection without additional fitting. We note that R_{bkg} is a single summary measure from the full ROC curve and differs from the common convention of quoting $1/\varepsilon_{\text{bkg}}$ at a fixed signal efficiency ε_{sig} (e.g. 50%); we adopt it because it follows directly from our scalar metrics.

4 Experiments

Training setup. We optimise with Ranger [42, 43] (RAdam combined with Lookahead), bf16 mixed precision, and DDP on four NVIDIA A100 GPUs. Learning rates are 10^{-3} for smaller models and 5×10^{-4} for `large`/`xlarge`. Batch sizes are 512 through `base`, 256 for `large`, and 128 for `xlarge`. Jobs checkpoint and requeue under SLURM.

Multi-epoch scaling. Figure 1a shows the headline multi-epoch `Herwig/full/pair` surface fit over 42 fully-trained runs ($\alpha = 0.21$, $\beta = 0.30$, $E = 0.24$, $R^2 = 0.974$; coefficients with uncertainties in Table 3), with observed points tracing the fitted surface closely.

Pythia and Herwig give statistically indistinguishable asymptotic accuracies ($\Delta\text{acc}_\infty = -0.002 \pm 0.013$, $P(\text{Pythia} > \text{Herwig}) = 0.41$; Table 3), so the headline applies to either generator; we use Herwig for its larger fully-trained grid (42 vs. 20 runs, giving $\sim 30\%$ tighter exponents).

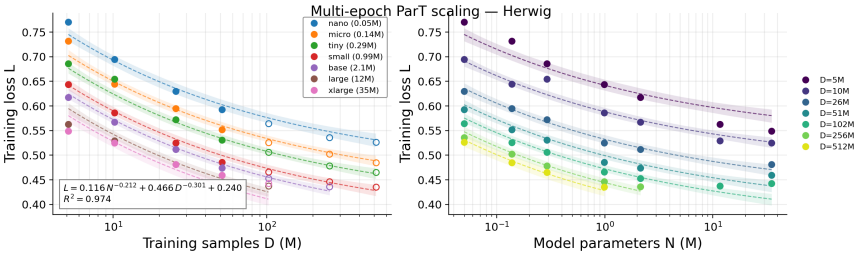
Single-epoch regime. In the 1-ep Pythia fit (Fig. 1b; Table 3), D equals the number of unique training jets by construction. The data exponent is larger and E is pinned to zero; the explored range (5–100M) is too narrow to resolve a floor, so the 1-ep curves capture local scaling in a data-limited regime—steeper in D and shallower in N than the multi-epoch fit, matching the "effective dataset size" picture of Vigl et al. [15].

Derived metrics from one loss fit. Composing the multi-epoch loss fit with the multi-epoch maps of Table 2 reproduces the observed validation accuracy, ROC AUC, and background rejection (Figure 2), with asymptotic ($N, D \to \infty$) values for Pythia of $\text{acc}_\infty \approx 0.895$ and $\text{ROC AUC}_\infty \approx 0.992$ (background rejection $R_{\text{bkg}} \approx 130$). The single-epoch fit, which never reuses data, extrapolates higher (0.942 and 0.997), consistent with reused samples yielding diminishing returns.

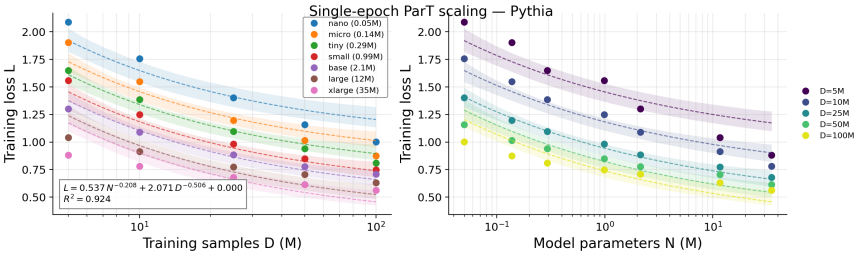
[1] Reference/URL to be added.

Table 3: Chinchilla fits to the training loss (frequentist $\pm 1\sigma$ from the fit covariance; a Bayesian NUTS fit agrees within 1σ, all $\hat{r} \leq 1.01$). Herwig is the headline (largest fully-trained grid); Vigl et al. [15] Set Transformer (SetT, Pythia) shown for reference.

Config	n	A	α	B	β	E	R^2
Herwig (multi)	42	0.116	0.21±0.04	0.466	0.30±0.05	0.240	0.974
Mixed (multi)	43	0.109	0.21±0.04	0.447	0.33±0.05	0.261	0.978
Pythia (multi)	20	0.086	0.25±0.06	0.481	0.40±0.08	0.279	0.987
Pythia (1-ep)	35	0.537	0.21±0.08	2.072	0.51±0.19	0.000	0.924
Vigl (SetT)	–	–	*0.44*	–	*0.22*	*0.32*	–



(a) Multi-epoch `Herwig/full/pair` (headline): $L = 0.116N^{-0.21} + 0.466D^{-0.30} + 0.240$ ($R^2 = 0.974$, $n = 42$). Open markers: data-reuse ($D > 100$M).



(b) Single-epoch `Pythia/full/pair`, where D is the number of unique jets; the data exponent is larger and E is pinned to zero.

Figure 1: ParT loss scaling in the two regimes (left: loss vs. D at fixed N; right: vs. N at fixed D). Shaded bands are the 94% Bayesian HDI (PyMC NUTS, $\hat{r} \leq 1.01$); coefficients with uncertainties in Table 3.

Pairwise-feature ablation. Disabling the pairwise features (`Herwig/full/nopair`, 46-cell grid) raises the model exponent ($\Delta\alpha = +0.110$) and the floor ($\Delta E = +0.088$) but leaves the data exponent unchanged ($\Delta\beta = +0.011$); the asymptotic background rejection falls $\sim 28\%$ (R^{∞}_{bkg} : $133 \rightarrow 96$). The pairwise features thus act as an architectural prior—each extra parameter must otherwise re-learn pair-like correlations—rather than a substitute for training data.

5 Discussion and conclusion

Comparison with baseline model. On Pythia (matching the SetT of Vigl et al. [15]), ParT has a lower model exponent ($\alpha = 0.25$ vs. 0.44) but a steeper data exponent ($\beta = 0.40$ vs. 0.22) and a lower floor ($E = 0.28$ vs. 0.32): its pairwise inductive bias makes it lean on data rather than parameters. The exponents are consistent across generators (Table 3), so this is not a generator artefact.

Conclusion and Future Work We presented a systematic scaling study of ParT on JetClass. ParT follows Chinchilla-style scaling ($\alpha \approx 0.21$, $\beta \approx 0.30$ on the headline Herwig grid), the steep data exponent indicating strong data efficiency, with fitted exponents consistent across Pythia, Herwig, and their mixture within uncertainty. Removing the pairwise features raises the model exponent and lowers the asymptotic background rejection by $\sim 28\%$, identifying them as an architectural prior rather than a substitute for data. Composed with simple per-regime $\log(1 - m)$ linear maps, a single cross-entropy fit also predicts accuracy, ROC AUC, and background rejection. Future work includes alternative inputs—e.g. a pure particle-level set (only per-particle kinematic four-vectors, no engineered identification or pairwise features)—and larger-scale pairwise-feature variations.

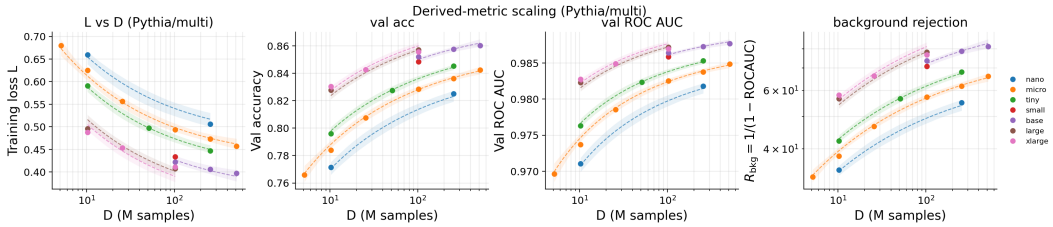


Figure 2: Derived-metric scaling (Pythia, multi-epoch) from composing the cross-entropy loss fit with the linear maps: loss, validation accuracy, validation ROC AUC, and background rejection vs. data budget D.

References

[1] Lyndon Evans and Philip Bryant. The large hadron collider. *Journal of Instrumentation*, 3:S08001, 2008.

[2] et al. Albertsson, Kim. Machine learning in high energy physics community white paper. *Journal of Physics: Conference Series*, 1085(2):022008, 2018.

[3] Roman Kogler et al. Jet substructure at the large hadron collider: Experimental review. *Reviews of Modern Physics*, 91(4):045003, 2019.

[4] Huilin Qu and Loukas Gouskos. Particlenet: Jet tagging via particle clouds. *Physical Review D*, 101(5):056019, 2020.

[5] Huilin Qu, Congqiao Li, and Sitian Qian. Particle transformer for jet tagging. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18281–18292. PMLR, 2022.

[6] Joschka Birk, Anna Hallin, and Gregor Kasieczka. OmniJet-α: the first cross-task foundation model for particle physics. *Mach. Learn. Sci. Tech.*, 5(3):035031, 2024.

[7] Huilin Qu, Congqiao Li, and Sitian Qian. Jetclass: A large-scale dataset for deep learning in jet physics. Zenodo, 2022.

[8] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[9] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[10] Jonathan S. Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales, 2019.

[11] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B. Brown, Prafulla Dhariwal, Scott Gray, Chris Hallacy, Benjamin Mann, Alec Radford, Aditya Ramesh, Nick Ryder, Daniel M. Ziegler, John Schulman, Dario Amodei, and Sam McCandlish. Scaling laws for autoregressive generative modeling, 2020.

[12] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md. Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically, 2017.

[13] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers, 2022.

[14] Armen Aghajanyan, Lili Yu, Alexis Conneau, Wei-Ning Hsu, Karen Hambardzumyan, Susan Zhang, Stephen Roller, Naman Goyal, Omer Levy, and Luke Zettlemoyer. Scaling laws for generative mixed-modal language models, 2023.

[15] Matthias Vigl, Nicole Hartman, Michael Kagan, and Lukas Heinrich. Neural scaling laws for boosted jet tagging. *arXiv preprint arXiv:2602.15781*, 2026.

[16] Torbjörn Sjöstrand, Stefan Ask, Jesper R. Christiansen, Richard Corke, Nishita Desai, Philip Ilten, Stephen Mrenna, Stefan Prestel, Christine O. Rasmussen, and Peter Z. Skands. An introduction to pythia 8.2. *Computer Physics Communications*, 191:159–177, 2015.

[17] Johannes Bellm and et al. Herwig 7.0/herwig++ 3.0 release note. *The European Physical Journal C*, 76(4):196, 2016.

[18] Patrick T Komiske, Eric M Metodiev, and Jesse Thaler. Energy flow polynomials: A complete linear basis for jet substructure. *Journal of High Energy Physics*, 2018(4):013, 2018.

[19] Gregor Kasieczka, Tilman Plehn, Anja Butter, Kyle Cranmer, and et al. The machine learning landscape of top taggers. *SciPost Physics*, 7(1), July 2019.

[20] Alexander Bogatskiy, Timothy Hoffman, David W. Miller, and Jan T. Offermann. Pelican: Permutation equivariant and lorentz invariant or covariant aggregator network for particle physics, 2022.

[21] ATLAS Collaboration. Transforming jet flavour tagging at atlas. *Nature Communications*, 17(1):541, 2026.

[22] Peter Battaglia et al. Interaction networks for learning about objects, relations and physics. In *NeurIPS*, 2016.

[23] Manzil Zaheer et al. Deep sets. In *NeurIPS*, 2017.

[24] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation, 2017.

[25] Jonas Spinner, Victor Bresó, Pim de Haan, Tilman Plehn, Jesse Thaler, and Johann Brehmer. Lorentz-equivariant geometric algebra transformers for high-energy physics, 2024.

[26] Anders Andreassen, Ilya Feige, Christopher Frye, and Matthew D. Schwartz. Junipr: a framework for unsupervised machine learning in particle physics. *The European Physical Journal C*, 79(2), February 2019.

[27] M. Erdmann, E. Geiser, Y. Rath, and M. Rieger. Lorentz boost networks: autonomous physics-inspired feature engineering. *Journal of Instrumentation*, 14(06):P06006–P06006, June 2019.

[28] Patrick T Komiske, Eric M Metodiev, and Jesse Thaler. Energy flow networks: Deep sets for particle jets. *Journal of High Energy Physics*, 2019(1):121, 2019.

[29] Gilles Louppe, Kyunghyun Cho, Cyril Becot, and Kyle Cranmer. Qcd-aware recursive neural networks for jet physics. *Journal of High Energy Physics*, 2019(1), January 2019.

[30] Shiqi Gong, Qi Meng, Jue Zhang, Huilin Qu, Congqiao Li, Sitian Qian, Weitao Du, Zhi-Ming Ma, and Tie-Yan Liu. An efficient lorentz equivariant graph neural network for jet tagging. *Journal of High Energy Physics*, 2022(7), July 2022.

[31] Josh Cogan, Michael Kagan, Emanuel Strauss, and Ariel Schwarztman. Jet-images: Computer vision inspired techniques for jet tagging. *Journal of High Energy Physics*, 2015(2):118, 2015.

[32] Daniel Guest, Julian Collado, Pierre Baldi, Shih-Chieh Hsu, Gregor Urban, and Daniel Whiteson. Jet flavor classification in high-energy physics with deep neural networks. *Physical Review D*, 94(11), December 2016.

[33] Barry Dillon, Gregor Kasieczka, Hans Olischlager, Tilman Plehn, Peter Sorrenson, and Lorenz Vogel. Symmetries, safety, and self-supervision. *SciPost Physics*, 12(6), June 2022.

[34] Tobias Golling, Lukas Heinrich, Michael Kagan, Samuel Klein, Matthew Leigh, Margarita Osadchy, and John Andrew Raine. Masked particle modeling on sets: Towards self-supervised high energy physics foundation models, 2024.

[35] Subash Katel, Haoyang Li, Zihan Zhao, Raghav Kansal, Farouk Mokhtar, and Javier Duarte. Learning symmetry-independent jet representations via jet-based joint embedding predictive architecture, 2024.

[36] Matthias Vigl, Nicole Hartman, and Lukas Heinrich. Finetuning foundation models for joint analysis optimization, 2024.

[37] Vinicius Mikuni and Benjamin Nachman. Solving key challenges in collider physics with foundation models, 2024.

[38] Matthew Leigh, Samuel Klein, François Charton, Tobias Golling, Lukas Heinrich, Michael Kagan, Inês Ochoa, and Margarita Osadchy. Is tokenization needed for masked particle modelling?, 2024.

[39] Jai Bardhan, Radhikesh Agrawal, Abhiram Tilak, Cyrin Neeraj, and Subhadip Mitra. Hep-jepa: A foundation model for collider physics using joint embedding predictive architecture, 2025.

[40] Wahid Bhimji, Chris Harris, Vinicius Mikuni, and Benjamin Nachman. Omnilearned: A foundation model framework for all tasks involving jet physics, 2025.

[41] Joschka Birk, Anna Hallin, Gregor Kasieczka, Nikol Madzharova, Ian Pang, and David Shih. Enhancing next token prediction based pre-training for jet foundation models, 2025.

[42] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond, 2020.

[43] Michael R. Zhang, James Lucas, Geoffrey Hinton, and Jimmy Ba. Lookahead optimizer: k steps forward, 1 step back, 2019.