Particle Transformer for Jet Tagging

Huilin Qu [1] **Congqiao Li** [2] **Sitian Qian** [2]

Abstract

Jet tagging is a critical yet challenging classification task in particle physics. While deep learning has transformed jet tagging and significantly improved performance, the lack of a large-scale public dataset impedes further enhancement. In this work, we present JetClass, a new comprehensive dataset for jet tagging. The JetClass dataset consists of 100 M jets, about two orders of magnitude larger than existing public datasets. A total of 10 types of jets are simulated, including several types unexplored for tagging so far. Based on the large dataset, we propose a new Transformer-based architecture for jet tagging, called Particle Transformer (ParT). By incorporating pairwise particle interactions in the attention mechanism, ParT achieves higher tagging performance than a plain Transformer and surpasses the previous state-of-the-art, ParticleNet, by a large margin. The pre-trained ParT models, once fine-tuned, also substantially enhance the performance on two widely adopted jet tagging benchmarks. The dataset, code and models are publicly available at `https://github.com/jet-universe/particle_transformer`.

1. Introduction

Machine learning has revolutionized how large-scale data samples are analyzed in particle physics and greatly increased the discovery potential for new fundamental laws of nature (Radovic et al., 2018). Specifically, deep learning has transformed how *jet tagging*, a critical classification task at high-energy particle colliders such as the CERN LHC, is performed, leading to a drastic improvement in its performance (Kogler et al., 2019; Larkoski et al., 2020).

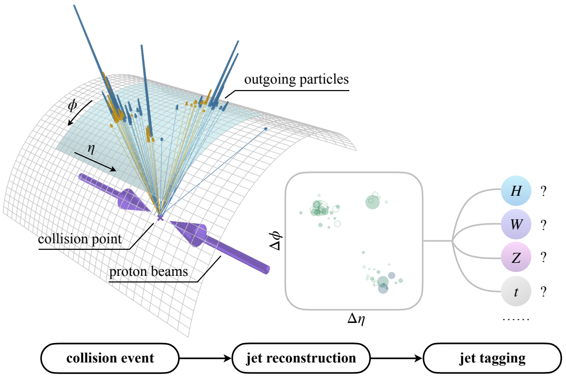


Figure 1. Illustration of jet tagging at the CERN LHC. High-energy proton-proton collisions at the LHC can produce new unstable particles that decay and yield a collimated spray of outgoing particles. These outgoing particles are measured by complex particle detector systems, and jets can be built (“reconstructed”) from these measured particles. The goal of jet tagging is to classify the jets and identify those arising from particles of high interest, e.g., the Higgs boson, the W or Z boson, or the top quark.

At the CERN LHC, two beams of protons are accelerated to nearly the speed of light and made to collide at a frequency of 40 million times per second (40 MHz). Such high-energy collisions can create new unstable particles, which then decay and produce sprays of outgoing particles. Complex detector systems, such as the general-purpose ATLAS (ATLAS Collaboration, 2008) and CMS (CMS Collaboration, 2008) detectors with $\mathcal{O}(100\,\text{M})$ individual sensors of various types, are used to measure the positions, trajectories, energies, and momenta of the outgoing particles. From these measurements, an *event* is reconstructed for each collision. The primary goal in the analysis of the collision data is to identify events involving novel physics processes, an example of which is the discovery of the Higgs boson (ATLAS Collaboration, 2012; CMS Collaboration, 2012).

A crucial step in the data analysis process is jet tagging. A *jet* refers to a collimated spray of outgoing particles. Jet tagging is the process of identifying the type of particle that initiates a jet. It is essentially a classification task that aims to distinguish jets arising from particles of interest, such as the Higgs boson or the top quark, from other less interesting types of jets. Jet tagging is a challenging task because the particle initiating a jet can radiate, and the

[1]CERN, Geneva, Switzerland [2]School of Physics, Peking University, Beijing, China. Correspondence to: Huilin Qu <huilin.qu@cern.ch>, Congqiao Li <licongqiao@pku.edu.cn>, Sitian Qian <stqian@pku.edu.cn>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

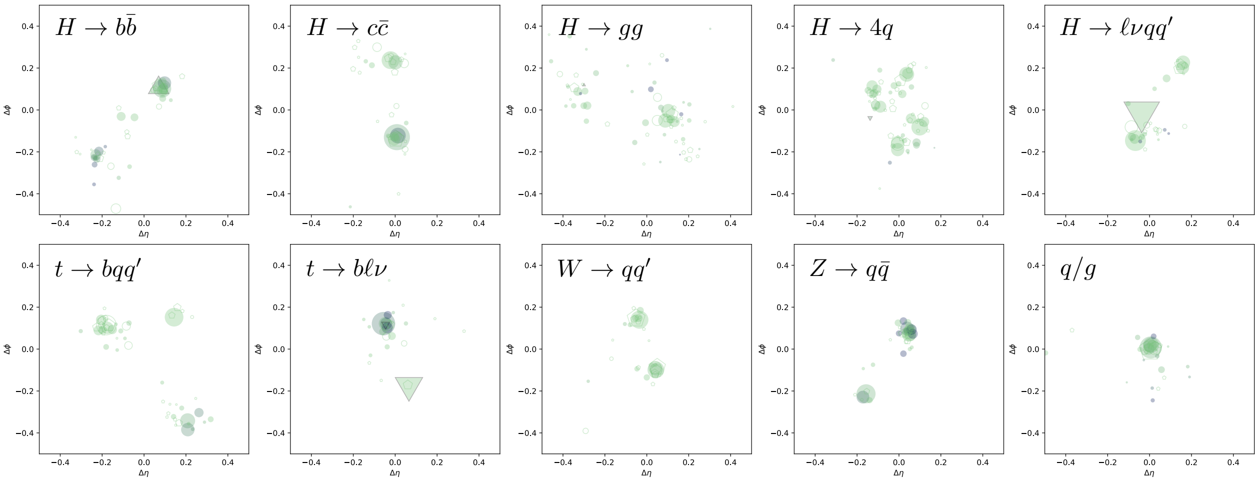


Figure 2. Examples of the 10 types of jets in the JetClass dataset, viewed as particle clouds. Each particle is displayed as a marker, with its coordinates corresponding to the flying direction of the particle, and its size proportional to the energy. The circles, triangles (upward- or downward-directed), and pentagons represent the particle types, which are hadrons, leptons (electrons or muons), and photons, respectively. The solid (hollow) markers stand for electrically charged (neutral) particles. The marker color reflects the displacement of the particle trajectory from the interaction point of the proton-proton collision, where a larger displacement results in more blue.

radiated particles further produce more particles, leading to a cascade of $\mathcal{O}(10)$ to $\mathcal{O}(100)$ particles at the end. The radiation also smears the characteristics of the initial particle and makes the identification very difficult.

Traditional approaches for jet tagging rely on hand-crafted features motivated by the principles of quantum chromodynamics (QCD), the theory governing the evolution of particles inside a jet. The rise of deep learning has led to a plethora of new approaches (Larkoski et al., 2020). The prevailing ones represent a jet as a *particle cloud*, i.e., an unordered and variable-sized set of the outgoing particles, as illustrated in Figure 1. Based on the particle cloud representation, ParticleNet (Qu & Gouskos, 2020) adapts the Dynamic Graph CNN architecture (Wang et al., 2019) and achieves substantial performance improvement on two representative jet tagging benchmarks. Since then, several new models (e.g., Mikuni & Canelli (2020; 2021); Shimmin (2021)) have been proposed, but no significant performance improvement has been reported so far. We deem the lack of a sufficiently large public dataset an impeding factor.

In this work, we advocate for JetClass, a new large and comprehensive dataset to advance deep learning for jet tagging. The JetClass dataset (Qu et al., 2022) consists of 100 M jets for training, about two orders of magnitude larger than existing public datasets. It also includes more types of jets, several of which have not been explored for tagging yet but are promising for future applications at the LHC.

Based on this dataset, we propose Particle Transformer (ParT), a new Transformer-based architecture for jet tagging. We demonstrate that Transformer architectures, together with a large dataset, can reach powerful performance on jet tagging. We introduce a small modification to the attention mechanism by incorporating a new term characterizing pairwise particle interactions. The resulting ParT achieves significantly higher performance than a plain Transformer and surpasses the previous state-of-the-art, ParticleNet, by a large margin. We also apply the pre-trained ParT models to two widely adopted jet tagging benchmarks with fine-tuning and observe a substantial gain on these tasks.

2. The JetClass Dataset

We provide an overview of the new JetClass dataset in this section. The dataset includes a total of 10 types of jets. Representative jets of each type are visualized as particle clouds in Figure 2. The jets in this dataset generally fall into two categories. The *background* jets are initiated by light quarks or gluons (q/g) and are ubiquitously produced at the LHC. The *signal* jets are those arising either from the top quarks (t), or from the W, Z or Higgs (H) bosons. For top quarks and Higgs bosons, we further consider their different decay modes as separate types, because the resulting jets have rather distinct characteristics and are often tagged individually. The use of jet tagging typically involves selecting one (or a few) specific type of signal jets with high confidence, and rejecting background jets as much as possible, since the background jets usually appear orders of magnitude more frequently than the targeted signal jets. Note that for several types of signal jets in this dataset, such as $H \to 4q$, $H \to \ell\nu qq'$, and $t \to b\ell\nu$, no dedicated methods have been developed so far to tag them. However, as we will demonstrate in Section 5.1, these types of jets can also be cleanly tagged with deep learning approaches, opening up new possible territories for jet tagging at the LHC.

Simulation setup. Jets in this dataset are simulated with standard Monte Carlo event generators used by LHC experiments. The production and decay of the top quarks and the W, Z and Higgs bosons are generated with MADGRAPH5_aMC@NLO (Alwall et al., 2014). We use PYTHIA (Sjöstrand et al., 2015) to evolve the produced particles, i.e., performing parton showering and hadronization, and produce the final outgoing particles[1]. To be close to realistic jets reconstructed at the ATLAS or CMS experiment, detector effects are simulated with DELPHES (de Favereau et al., 2014) using the CMS detector configuration provided in DELPHES. In addition, the impact parameters of electrically charged particles are smeared to match the resolution of the CMS tracking detector (CMS Collaboration, 2014). Jets are clustered from DELPHES E-Flow objects with the anti-k_{T} algorithm (Cacciari et al., 2008; 2012) using a distance parameter $R = 0.8$. Only jets with transverse momentum in 500–1000 GeV and pseudorapidity $|\eta| < 2$ are considered. For signal jets, only the "high-quality" ones that fully contain the decay products of initial particles are included[2].

Input features. The dataset provides all constituent particles of each jet as inputs for jet tagging. Note that the number of particles varies from jet to jet, typically between 10 and 100, with an average of 30–50 depending on the jet type. For each particle of a jet, three categories of features are provided:

- **Kinematics.** This includes the energy and momentum, described by the 4-vector (E, p_x, p_y, p_z) in units of GeV, which are the most fundamental quantities measured by a particle detector. All other kinematic variables can be computed from the 4-vectors.
- **Particle identification.** This includes the electric charge, with values of ± 1 (positively/negatively charged particles) and 0 (neural particles), and the particle identity determined by the detector systems. For the latter, a 5-class one-hot encoding is used to be consistent with current LHC experiments: charged hadron ($\pm 211, \pm 321, \pm 2212$), neutral hadron (0), electron (± 11), muon (± 13), and photon (22). The particle identification information is especially important for tagging jets involving a charged lepton, e.g., $H \to \ell\nu qq'$ and $t \to b\ell\nu$, as leptons can be almost unambiguously identified at the LHC.
- **Trajectory displacement.** This includes the measured values and errors of the transverse and longitudinal impact parameters of the particle trajectories in units of mm, in total 4 variables. These measurements are only available for electrically charged particles, and a value of 0 is used for neutral particles. The trajectory displacement information is critical for tagging jets involving a bottom (b) or charm (c) quark (CMS Collaboration, 2020b), such as $H \to b\bar{b}$, $H \to c\bar{c}$, $t \to bqq'$, etc., but is missing from most of the existing datasets.

Training, validation and test sets. The training set consists of 100 M jets in total, equally distributed in the 10 classes. An additional set of 500 k jets per class (in total 5 M) is intended for model validation. For the evaluation of performance, a separate test set with 2 M jets in each class (in total 20 M) is provided.

Evaluation metrics. To thoroughly evaluate the performance of deep learning models on this dataset, we advocate for a series of metrics. Since jet tagging on this dataset is naturally framed as a multi-class classification task, two common metrics, i.e., the accuracy and the area under the ROC curve (AUC)[3] are adopted to quantify the overall performance. In addition, we propose the *background rejection* (i.e., the inverse of the false positive rate) at a certain signal efficiency (i.e., the true positive rate, TPR) of $X\%$, i.e.,

$$\text{Rej}_{X\%} \equiv 1/\text{FPR at TPR} = X\%, \tag{1}$$

for each type of signal jets. By default, the q/g jets are considered as the background, as is the case in most LHC data analyses, and each of the other 9 types of jets can be considered as the signal. The signal efficiency (TPR) for each signal type is chosen to be representative of actual usages at the LHC experiments and is typically 50%. It is increased to 99% (99.5%) for $H \to \ell\nu qq'$ ($t \to b\ell\nu$), as these types of jets have more distinct characteristics and can be more easily separated from q/g jets. Since the definition of the Rej_X metric involves only two classes, i.e., the signal class under consideration (S) and the background class (B), the TPR and FPR are evaluated using a two-class score,

$$\text{score}_{S\text{vs}B} \equiv \frac{\text{score}(S)}{\text{score}(S) + \text{score}(B)}, \tag{2}$$

where score(S) and score(B) are the softmax outputs for class S and B, respectively, to achieve optimal performance for S vs B separation. This is aligned with the convention adopted by the CMS experiment (CMS Collaboration, 2020b). Note that the background rejection metric, although rarely used in vision or language tasks, is actually a standard metric for jet tagging because it is directly related to the discovery potential at the LHC experiments. A factor

[1] We include multiple parton interactions but omit pileup interactions in the simulation.

[2] We require all the quarks (q) and charged leptons (electrons or muons, denoted ℓ) from the decay of the top quark or the W, Z or Higgs boson satisfy $\Delta R(\text{jet}, q/\ell) < 0.8$, where $\Delta R(a, b) \equiv \sqrt{(\eta_a - \eta_b)^2 + (\phi_a - \phi_b)^2}$, in which η (ϕ) is the pseudorapidity (azimuthal angle) of the momentum of the jet or the particle.

[3] The AUC can be calculated using `roc_auc_score` in scikit-learn with `average='macro'` and `multi_class='ovo'`.

of two increase in background rejection can lead to about 40% increase in the discovery potential, which would otherwise require a dataset of twice the size, or in other words, doubling the running time of the LHC.

3. Related Work

Jet tagging with deep learning. Deep learning approaches have been proposed extensively to improve jet tagging. Previous models handle jets with different representations, e.g., images (de Oliveira et al., 2016), sequences (Guest et al., 2016), trees (Louppe et al., 2019), graphs (Henrion et al., 2017), with corresponding deep learning architectures such as 2D CNNs, recurrent or recursive networks, and graph neural networks. More recently, the particle cloud representation (Komiske et al., 2019b; Qu & Gouskos, 2020), analogous to point clouds, which treats a jet as a permutation-invariant set of particles as visualized in Figure 2, has been proposed. The Deep Sets (Zaheer et al., 2017) and Dynamic Graph CNN (Wang et al., 2019) architectures are adapted for jet tagging, resulting in the Energy Flow Network (Komiske et al., 2019b) and the state-of-the-art, ParticleNet (Qu & Gouskos, 2020), respectively. Since then, particle clouds have become the prevailing representation of jets and more architectures based on GAPNet (Chen et al., 2021; Mikuni & Canelli, 2020), the Point Cloud Transformer (Guo et al., 2021; Mikuni & Canelli, 2021) have been studied, but no significant performance improvement over ParticleNet has been reported. Lately, researches have been focused more on incorporating inductive biases from physics principles in the architecture design, such as the usage of the Lund jet plane (Dreyer et al., 2018; Dreyer & Qu, 2021; Dreyer et al., 2021; 2022), the Lorentz group symmetry (Bogatskiy et al., 2020; Gong et al., 2022), and the rotational symmetry (Shimmin, 2021; Dillon et al., 2021).

Deep-learning-based jet tagging algorithms have been widely adopted in real-world data analysis at the LHC. For example, the CMS Collaboration develops the DeepAK8 (CMS Collaboration, 2020b) algorithm to tag jets arising from the top quark or the Higgs, W, or Z boson, using a 1D CNN following the ResNet (He et al., 2016) architecture, and a significant increase in the discovery potential for new heavy particles has been achieved (CMS Collaboration, 2021; 2022a). Moreover, using ParticleNet, CMS achieves the first observation of Z boson decay to a pair of charm quarks at a hadron collider and obtains the most stringent constraint on $H \to c\bar{c}$ decay (CMS Collaboration, 2022c). ParticleNet is also used by CMS to probe the quartic interaction between the Higgs and vector bosons, indirectly confirming its existence for the first time (CMS Collaboration, 2022b). Clearly, advances in jet tagging play a vital role in accelerating our understanding of elementary particles, the fundamental building blocks of nature.

Jet tagging datasets. A number of datasets have been published so far to study jet tagging:

- **Top quark tagging dataset** (Kasieczka et al., 2019) proposed in Butter et al. (2019), consisting of 2 M jets in 2 types ($t \to bqq'$ and q/g) and providing only the kinematic information.
- **Quark-gluon tagging dataset** (Komiske et al., 2019a) proposed in Komiske et al. (2019b), consisting of 2 M jets in 2 types (quark and gluon), and providing both the kinematic and particle identification information.
- **Higgs boson tagging dataset** (Duarte, 2019; Chen et al., 2022), containing 3.9 M $H \to b\bar{b}$ jets and 1.9 M q/g jets, with all three categories of information.
- **JetNet dataset** (Kansal et al., 2021b) proposed in Kansal et al. (2021a), containing $\approx$500 k jets in 3 types: gluon, light quark, and top quark, and providing only the kinematic information.
- **A multiclass dataset** (Pierini et al., 2020) proposed in Moreno et al. (2020), with 880 k jets in 5 classes: light quark, gluon, W boson, Z boson and top quark and providing only the kinematic information.

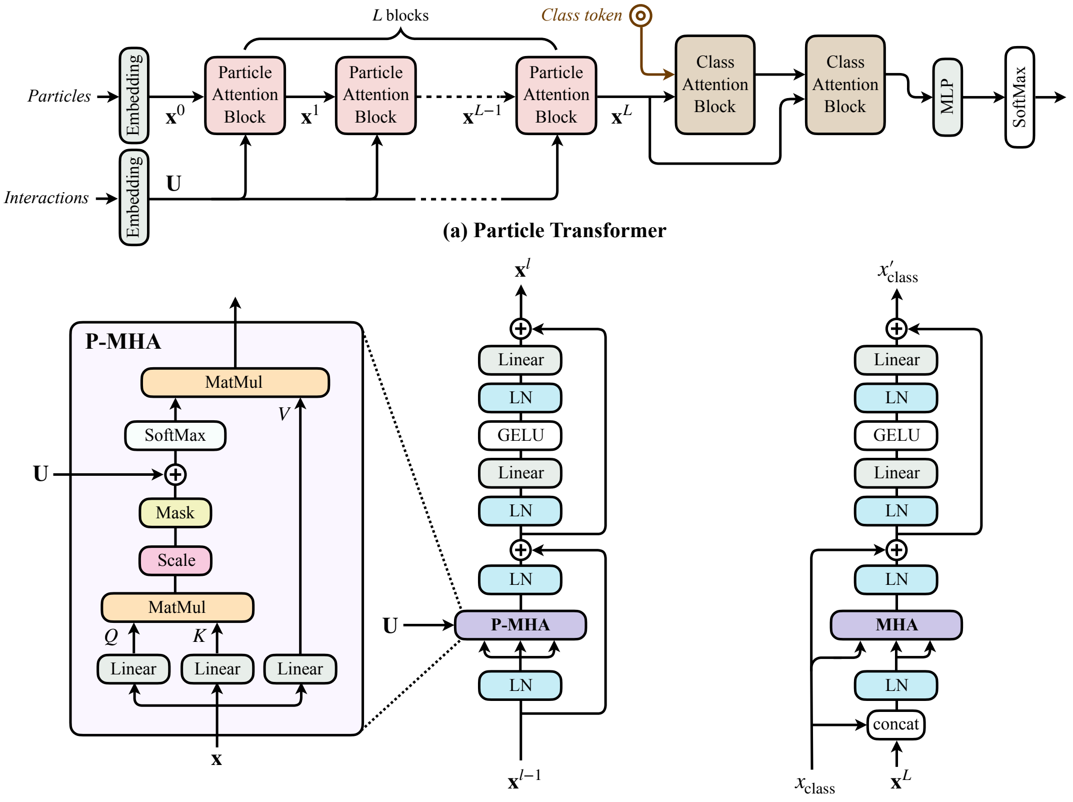
Compared with existing datasets, the JetClass dataset is not only substantially larger in size, but also more inclusive in terms of the types of jets contained.

Transformers. Recent years have witnessed the enormous success of Transformer models. Starting from natural language processing and then spreading to computer vision, the original Transformer (Vaswani et al., 2017), as well as its variants, e.g., BERT (Devlin et al., 2019), ViT (Dosovitskiy et al., 2021) and Swin-Transformer (Liu et al., 2021), have refreshed the performance records in various tasks, demonstrating the power of Transformer as a universal architecture. Transformers, and the attention mechanism at its core, have proved to be powerful for fundamental scientific problems as well. For example, AlphaFold2 (Jumper et al., 2021), which reaches the state-of-the-art performance in protein structure prediction, employs the attention mechanism. In particular, adding a pair bias, derived from pairwise features, to the self attention helps improve the model explainability.

4. Model Architecture

Together with the JetClass dataset, we propose the Particle Transformer (ParT) as a new baseline for jet tagging. An overview of the ParT architecture is presented in Figure 3(a). For a jet with N particles, ParT makes use of two sets of inputs: the *particle* input includes a list of C features for every particle and forms an array of a shape (N, C); the *interaction*[4] input is a matrix of C' features for every pair

[4] The term *interaction* here refers to any feature involving a pair of particles, which may or may not be related to the physical forces between them.



(b) Particle Attention Block **(c) Class Attention Block**

Figure 3. The architecture of (a) Particle Transformer (b) Particle Attention Block (c) Class Attention Block.

of particles, in a shape (N, N, C'). The particle and interaction inputs are each followed by an MLP to project them to a d- and d'-dimensional embedding, $\mathbf{x}^0 \in \mathbb{R}^{N\times d}$ and $\mathbf{U} \in \mathbb{R}^{N\times N\times d'}$, respectively. Unlike Transformers for NLP and vision, we do not add any ad-hoc positional encodings, as the particles in a jet are permutation invariant. The spatial information (i.e., the flying direction of each particle) is directly included in the particle inputs. We feed the particle embedding $\mathbf{x}^0$ into a stack of L particle attention blocks to produce new embeddings, $\mathbf{x}^1, ..., \mathbf{x}^L$ via multi-head self attention. The interaction matrix $\mathbf{U}$ is used to augment the scaled dot-product attention by adding it as a bias to the pre-softmax attention weights. The same $\mathbf{U}$ is used for all the particle attention blocks. After that, the last particle embedding $\mathbf{x}^L$ is fed into two class attention blocks, and a global class token x_{class} is used to extract information for jet classification via attention to all the particles, following the CaiT approach (Touvron et al., 2021). The class token is passed to a single-layer MLP, followed by softmax, to produce the final classification scores.

Remark. ParT can also be viewed as a graph neural network on a fully-connected graph, in which each node corresponds to a particle, and the interactions are the edge features.

Particle interaction features. While the ParT architecture is designed to be able to process any kinds of pairwise interaction features, for this paper we only consider a specific scenario in which the interaction features are derived from the energy-momentum 4-vector, $p = (E, p_x, p_y, p_z)$, of each particle. This is the most general case for jet tagging, as the particle 4-vectors are available in every jet tagging task. Specifically, for a pair of particles a, b with 4-vectors p_a, p_b, we calculate the following 4 features:

$$\begin{aligned}\Delta &= \sqrt{(y_a - y_b)^2 + (\phi_a - \phi_b)^2},\\ k_{\mathrm{T}} &= \min(p_{\mathrm{T},a}, p_{\mathrm{T},b})\Delta,\\ z &= \min(p_{\mathrm{T},a}, p_{\mathrm{T},b})/(p_{\mathrm{T},a} + p_{\mathrm{T},b}),\\ m^2 &= (E_a + E_b)^2 - \|\mathbf{p}_a + \mathbf{p}_b\|^2,\end{aligned} \tag{3}$$

where y_i is the rapidity, ϕ_i is the azimuthal angle, $p_{\mathrm{T},i} = (p_{x,i}^2 + p_{y,i}^2)^{1/2}$ is the transverse momentum, and $\mathbf{p}_i = (p_{x,i}, p_{y,i}, p_{z,i})$ is the momentum 3-vector and $\|\cdot\|$ is the norm, for $i = a$, b. Since these variables typically have a long-tail distribution, we take the logarithm and use $(\ln\Delta, \ln k_{\mathrm{T}}, \ln z, \ln m^2)$ as the interaction features for each particle pair. The choice of this set of features is motivated by Dreyer & Qu (2021).

Particle attention block. A key component of ParT is the particle attention block. As illustrated in Figure 3(b), the particle attention block consists of two stages. The first stage includes a multi-head attention (MHA) module with a LayerNorm (LN) layer both before and afterwards. The second stage is a 2-layer MLP, with an LN before each linear layer and GELU nonlinearity in between. Residual connections are added after each stage. The overall block structure is based on NormFormer (Shleifer et al., 2021), however, we replace the standard MHA with P-MHA, an augmented version that can also exploit the pairwise particle interactions directly. The P-MHA is computed as

$$\text{P-MHA}(Q, K, V) = \text{SoftMax}(QK^T/\sqrt{d_k} + \mathbf{U})V, \quad (4)$$

where Q, K and V are linear projections of the particle embedding $\mathbf{x}^l$. Essentially, we add the interaction matrix $\mathbf{U}$ to the pre-softmax attention weights. This allows P-MHA to incorporate particle interaction features designed from physics principles and modify the dot-product attention weights, thus increasing the expressiveness of the attention mechanism.

Class attention block. As illustrated in Figure 3(c), the class attention block has a similar structure as the particle attention block. However, unlike in the particle attention block where we compute the self attention between particles, here we compute the attention between a global class token x_{class} and all the particles using the standard MHA. Specifically, the inputs to the MHA are

$$\begin{aligned} Q &= W_q x_{\text{class}} + b_q, \\ K &= W_k \mathbf{z} + b_k, \\ V &= W_v \mathbf{z} + b_v, \end{aligned} \quad (5)$$

where $\mathbf{z} = [x_{\text{class}}, \mathbf{x}^L]$ is the concatenation of the class token and the particle embedding after the last particle attention block, $\mathbf{x}^L$.

Implementation. We implement the ParT model in PyTorch (Paszke et al., 2019). Specifically, the P-MHA is implemented using the PyTorch's `MultiheadAttention` by providing the interaction matrix $\mathbf{U}$ as the `attn_mask` input. The baseline ParT model has a total of $L = 8$ particle attention blocks and 2 class attention blocks. It uses a particle embedding of a dimension $d = 128$, encoded from the input particle features using a 3-layer MLP with (128, 512, 128) nodes each layer with GELU nonlinearity, and LN is used in between for normalization. The interaction input features are encoded using a 4-layer pointwise 1D convolution with (64, 64, 64, 16) channels with GELU nonlinearity and batch normalization in between to yield a $d' = 16$ dimensional interaction matrix. The P-MHA (MHA) in the particle (class) attention blocks all have 8 heads, with a query dimension $d' = 16$ for each head, and an expansion factor of 4 for the MLP. We use a dropout of 0.1 for all particle attention blocks, and no dropout for the class attention block. The choice of hyperparameters provides a reasonable baseline but is not extensively optimized.

5. Experiments

We conduct experiments on the new JetClass dataset and show the results in Section 5.1. The pre-trained ParT models are also applied to two existing datasets with fine-tuning, and the performance is compared to previous state-of-the-arts in Section 5.2.

5.1. Experiments on JetClass Dataset

Setup. For experiments on the JetClass dataset, we use the full set of particle features, including kinematics, particle identification, and trajectory displacement, as inputs. The full list of 17 features for each particle is summarized in Table 2. In addition, the 4 interaction features introduced in Equation (3) are also used for the ParT model. The training is performed on the full training set of 100 M jets. We employ the Lookahead optimizer (Zhang et al., 2019) with $k = 6$ and $\alpha = 0.5$ to minimize the cross-entropy loss, and the inner optimizer is RAdam (Liu et al., 2020) with $\beta_1 = 0.95$, $\beta_2 = 0.999$, and $\epsilon = 10^{-5}$. A batch size of 512 and an initial learning rate (LR) of 0.001 are used. No weight decay is applied. We train for a total of 1 M iterations, amounting to around 5 epochs over the full training set. The LR remains constant for the first 70% of the iterations, and then decays exponentially, at an interval of every 20 k iterations, down to 1% of the initial value at the end of the training. Performance of the model is evaluated every 20 k iterations on the validation set and a model checkpoint is saved. The checkpoint with the highest accuracy on the validation set is used to evaluate the final performance on the test set.

Baselines. We compare the performance of ParT with 3 baseline models: the PFN (Komiske et al., 2019b) architecture based on Deep Sets (Zaheer et al., 2017), the P-CNN architecture used by the DeepAK8 algorithm of the CMS experiment (CMS Collaboration, 2020b), and the state-of-the-art ParticleNet architecture (Qu & Gouskos, 2020) adapted from DGCNN (Wang et al., 2019). All the models are trained end-to-end on the JetClass dataset for the same number of effective epochs for a direct comparison. For ParticleNet, we directly use the existing PyTorch implementation. For PFN and P-CNN, we re-implement them in PyTorch and verify that the published results are reproduced. The optimizer and LR schedule remain the same as in the training of ParT. The (batch size, LR) combination is re-optimized and chosen to be (512, 0.01) for ParticleNet and (4096, 0.02) for PFN and P-CNN.

Results. Performance on the JetClass dataset is evaluated using the metrics described in Section 2, and the results are summarized in Table 1. The proposed ParT architecture achieves the best performance on every metric, and outperforms the existing state-of-the-art, ParticleNet, by a large margin. The overall accuracy is increased by 1.7% com-

Table 1. Jet tagging performance on the JetClass dataset. ParT is compared to PFN (Komiske et al., 2019b), P-CNN (CMS Collaboration, 2020b) and the state-of-the-art ParticleNet (Qu & Gouskos, 2020). For all the metrics, a higher value indicates better performance. The ParT architecture using plain MHAs instead of P-MHAs, labelled as ParT (plain), is also shown for comparison.

	All classes		$H \to b\bar{b}$	$H \to c\bar{c}$	$H \to gg$	$H \to 4q$	$H \to \ell\nu qq'$	$t \to bqq'$	$t \to b\ell\nu$	$W \to qq'$	$Z \to q\bar{q}$
	Accuracy	AUC	$\text{Rej}_{50\%}$	$\text{Rej}_{50\%}$	$\text{Rej}_{50\%}$	$\text{Rej}_{50\%}$	$\text{Rej}_{99\%}$	$\text{Rej}_{50\%}$	$\text{Rej}_{99.5\%}$	$\text{Rej}_{50\%}$	$\text{Rej}_{50\%}$
PFN	0.772	0.9714	2924	841	75	198	265	797	721	189	159
P-CNN	0.809	0.9789	4890	1276	88	474	947	2907	2304	241	204
ParticleNet	0.844	0.9849	7634	2475	104	954	3339	10526	11173	347	283
ParT	**0.861**	**0.9877**	**10638**	**4149**	**123**	**1864**	**5479**	**32787**	**15873**	**543**	**402**
ParT (plain)	0.849	0.9859	9569	2911	112	1185	3868	17699	12987	384	311

Table 2. Particle input features used for jet tagging on the JetClass, the top quark tagging (Top) and the quark gluon tagging (QG) datasets. For QG, we consider two scenarios: QG_{exp} is restricted to use only the 5-class experimentally realistic particle identification information, while QG_{full} uses the full set of particle identification information in the dataset and further distinguish between different types of charged hadrons and neutral hadrons.

Category	Variable	Definition	JetClass	Top	QG_{exp}	QG_{full}
Kinematics	$\Delta\eta$	difference in pseudorapidity η between the particle and the jet axis	✓	✓	✓	✓
	$\Delta\phi$	difference in azimuthal angle ϕ between the particle and the jet axis	✓	✓	✓	✓
	$\log p_{\mathrm{T}}$	logarithm of the particle's transverse momentum p_{T}	✓	✓	✓	✓
	$\log E$	logarithm of the particle's energy	✓	✓	✓	✓
	$\log \frac{p_{\mathrm{T}}}{p_{\mathrm{T}}(\text{jet})}$	logarithm of the particle's p_{T} relative to the jet p_{T}	✓	✓	✓	✓
	$\log \frac{E}{E(\text{jet})}$	logarithm of the particle's energy relative to the jet energy	✓	✓	✓	✓
	ΔR	angular separation between the particle and the jet axis ($\sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$)	✓	✓	✓	✓
Particle identification	`charge`	electric charge of the particle	✓	—	✓	✓
	`Electron`	if the particle is an electron (`|pid|==11`)	✓	—	✓	✓
	`Muon`	if the particle is an muon (`|pid|==13`)	✓	—	✓	✓
	`Photon`	if the particle is an photon (`pid==22`)	✓	—	✓	✓
	`CH`	if the particle is an charged hadron (`|pid|==211 or 321 or 2212`)	✓	—	✓	✓[a]
	`NH`	if the particle is an neutral hadron (`|pid|==130 or 2112 or 0`)	✓	—	✓	✓[b]
Trajectory displacement	$\tanh d_0$	hyperbolic tangent of the transverse impact parameter value	✓	—	—	—
	$\tanh d_z$	hyperbolic tangent of the longitudinal impact parameter value	✓	—	—	—
	σ_{d_0}	error of the measured transverse impact parameter	✓	—	—	—
	σ_{d_z}	error of the measured longitudinal impact parameter	✓	—	—	—

[a] `(|pid|==211) + (|pid|==321)*0.5 + (|pid|==2212)*0.2`

[b] `(|pid|==130) + (|pid|==2112)*0.2.`

pared to ParticleNet. Moreover, for the physics-oriented metric, the background rejection, ParT improves over ParticleNet by a factor of 3 for $t \to bqq'$, a factor of 2 for $H \to 4q$, and about 70% for $H \to c\bar{c}$. It is also clear that, the earlier PFN and P-CNN models lag substantially behind ParticleNet and ParT on this large dataset, amounting to up to an order of magnitude difference in background rejection. The large improvement of ParT is likely to lead to a significant jump in the discovery potential for related physics searches at the LHC.

Another observation is that there is a large variation in tagging performance between signals of different types. The best separation against the background q/g jets is achieved for $t \to b\ell\nu$ and $H \to \ell\nu qq'$ signals – with the powerful ParT model, these two can be selected almost perfectly, i.e., at an efficiency of more than 99% with nearly no contamination from background jets. This opens up new territory for jet tagging at the LHC, as these types of jets have not been exploited for tagging so far.

Effectiveness of P-MHA. To quantify the effectiveness of the P-MHA introduced in ParT, we carry out an ablation study by replacing the P-MHA with a standard MHA, the resulting architecture is then a plain Transformer and therefore denoted as ParT (plain). We train ParT (plain) with the same procedure as the full ParT and the performance is shown in Table 1. A drop of 1.2% in accuracy is observed compared to the full ParT, and the background rejection is reduced by 20–30% for most signals. Note that, replacing P-MHA with plain MHA implies that the particle interaction input is discarded completely, but this does not lead to any reduction of information content, as the interaction features defined in Equation (3) are derived purely from the energy-momentum 4-vectors, which are already used as particle features via the 7 kinematic variables presented in Table 2. Therefore, the improvement of ParT over a plain Transformer indeed arise from an efficient exploitation of the particle kinematic information using the P-MHA.

Table 3. Impacts of the training dataset size. Entries in bold correspond to the training using the full 100 M training dataset.

	All classes		$H \to b\bar{b}$	$H \to c\bar{c}$	$H \to gg$	$H \to 4q$	$H \to \ell\nu qq'$	$t \to bqq'$	$t \to b\ell\nu$	$W \to qq'$	$Z \to q\bar{q}$
	Accuracy	AUC	Rej$_{50\%}$	Rej$_{50\%}$	Rej$_{50\%}$	Rej$_{50\%}$	Rej$_{99\%}$	Rej$_{50\%}$	Rej$_{99.5\%}$	Rej$_{50\%}$	Rej$_{50\%}$
ParticleNet (2 M)	0.828	0.9820	5540	1681	90	662	1654	4049	4673	260	215
ParticleNet (10 M)	0.837	0.9837	5848	2070	96	770	2350	5495	6803	307	253
ParticleNet (100 M)	0.844	0.9849	7634	2475	104	954	3339	10526	11173	347	283
ParT (2 M)	0.836	0.9834	5587	1982	93	761	1609	6061	4474	307	236
ParT (10 M)	0.850	0.9860	8734	3040	110	1274	3257	12579	8969	431	324
ParT (100 M)	0.861	0.9877	10638	4149	123	1864	5479	32787	15873	543	402

Table 4. Number of trainable parameters and FLOPs.

	Accuracy	# params	FLOPs
PFN	0.772	86.1 k	4.62 M
P-CNN	0.809	354 k	15.5 M
ParticleNet	0.844	370 k	540 M
ParT	**0.861**	2.14 M	340 M
ParT (plain)	0.849	2.13 M	260 M

Impacts of the training dataset size. To evaluate the impacts of the training dataset size on the jet tagging performance, we perform additional trainings using only 2% and 10% of the JETCLASS dataset. For the former, the training is performed for only 100 k iterations, as it is already converged by then. For the latter, the training still lasts for 1 M iterations, although very little gain is observed compared to the training with only 100 k iterations. No overfitting is found in either case. The results are summarized in Table 3. For the ParticleNet model, a drop of 0.7% in accuracy is observed when the training dataset size is reduced to 10 M, and the drop in accuracy increases to 1.6% when only 2 M jets are used in the training. For the ParT model, the impact is even larger, the degradation in accuracy becomes 1.1% and 2.5% when the training dataset is reduced to 10% and 2%, respectively.

Model complexity. Table 4 compares the model complexity of ParT with the baselines. While the number of trainable parameters is increased by more than 5× compared to ParticleNet, the number of floating point operations (FLOPs) is actually 40% lower. We also observe that the FLOPs of ParT are 30% higher than ParT (plain), which mostly comes from the encoding of the pairwise features, because the computational cost there scales quadratically with the number of particles in a jet.

5.2. Fine-Tuning for Other Datasets

Top quark tagging dataset. The top quark tagging benchmark (Butter et al., 2019) provides a dataset of 2 M (1.2/0.4/0.4 M for train/validation/test) jets in two classes, $t \to bqq'$ (signal) and q/g (background). Only kinematic features, i.e., the energy-momentum 4-vectors, are provided. Therefore, we pre-train a ParT model on the JETCLASS dataset using only the kinematic features, and then fine-tune it on the top quark tagging dataset. The particle input features are the 7 kinematic features listed in Table 2, the same as used by ParticleNet. The JETCLASS pre-training follows the same setup as described in Section 5.1. For the fine-tuning, we replace the last MLP with a new randomly-initialized MLP with 2 output nodes, and then fine-tune all the weights on the top tagging dataset for 20 epochs. A smaller LR of 0.0001 is used for the pre-trained weights, while a larger LR of 0.005 is used to update the randomly-initialized weights of the MLP. The LR remains constant across the full training, with a weight decay of 0.01. We run a total of 9 experiments, starting from the same pre-trained model but different random initializations of the replaced MLP, and report the performance of the model with median accuracy and the spread across the 9 trainings, following the procedure used by ParticleNet. For comparison, we also train ParT from scratch on this dataset for 20 epochs, using a start LR of 0.001, a schedule that decays the LR to 1% in the last 30% of the epochs, and a weight decay of 0.01. Both results are presented in Table 5. The pre-trained ParT achieves a significant improvement over the existing baselines, increasing Rej$_{30\%}$ by 70% compared to ParticleNet, and by 26% compared to the best-performing model on this dataset, LorentzNet. On the other hand, the ParT model trained from scratch only reaches similar performance as ParticleNet. We also investigate a similar pre-training and fine-tuning procedure using the ParticleNet model, but only a small improvement is observed compared to the training from scratch, due to the limited capacity of the ParticleNet model.

Quark-gluon tagging dataset. We also benchmark ParT on the quark-gluon tagging dataset (Komiske et al., 2019a) proposed in Komiske et al. (2019b), the target of which is to separate jets initiated by quarks (signal) from those by gluons (background). This dataset also consists of 2 M jets, with a recommended train/validation/test splitting of 1.6/0.2/0.2 M. It provides not only the kinematic features, but also particle identification information. We consider two scenarios in the usage of the particle identification information. In the "exp" scenario, we restrict the information to only 5 classes and do not attempt to separate electrically charged (and neural) hadrons of different types, which is the procedure adopted by ParticleNet, and also prescribed by

Table 5. Comparison between ParT and existing models on the top quark tagging dataset. ParT refers to the model trained from scratch on this dataset. ParticleNet-f.t. and ParT-f.t. denote the corresponding models pre-trained on JetClass and fine-tuned on this dataset. Results for other models are quoted from their published results: P-CNN and ParticleNet (Qu & Gouskos, 2020), PFN (Komiske et al., 2019b), JEDI-net (Moreno et al., 2020), PCT (Mikuni & Canelli, 2021), LGN (Bogatskiy et al., 2020), rPCN (Shimmin, 2021), and LorentzNet (Gong et al., 2022).

	Accuracy	AUC	$\text{Rej}_{50\%}$	$\text{Rej}_{30\%}$
P-CNN	0.930	0.9803	201 ± 4	759 ± 24
PFN	—	0.9819	247 ± 3	888 ± 17
ParticleNet	0.940	0.9858	397 ± 7	1615 ± 93
JEDI-net (w/ $\sum O$)	0.930	0.9807	—	774.6
PCT	0.940	0.9855	392 ± 7	1533 ± 101
LGN	0.929	0.964	—	435 ± 95
rPCN	—	0.9845	364 ± 9	1642 ± 93
LorentzNet	0.942	0.9868	498 ± 18	2195 ± 173
ParT	0.940	0.9858	413 ± 16	1602 ± 81
ParticleNet-f.t.	0.942	0.9866	487 ± 9	1771 ± 80
ParT-f.t.	**0.944**	**0.9877**	$\mathbf{691 \pm 15}$	$\mathbf{2766 \pm 130}$

Table 6. Comparison between ParT and existing models on the quark-gluon tagging dataset. ParT refers to the model trained from scratch on this dataset. ParticleNet-f.t. and ParT-f.t. denote the corresponding models pre-trained on JetClass and fine-tuned on this dataset. Results for other models are quoted from their published results: P-CNN and ParticleNet (Qu & Gouskos, 2020), PFN (Komiske et al., 2019b), ABCNet (Mikuni & Canelli, 2020), PCT (Mikuni & Canelli, 2021), rPCN (Shimmin, 2021), and LorentzNet (Gong et al., 2022). The subscript “exp” and “full” distinguish models using partial or full particle identification information.

	Accuracy	AUC	$\text{Rej}_{50\%}$	$\text{Rej}_{30\%}$
$\text{P-CNN}_{\text{exp}}$	0.827	0.9002	34.7	91.0
PFN_{exp}	—	0.9005	34.7 ± 0.4	—
$\text{ParticleNet}_{\text{exp}}$	0.840	0.9116	39.8 ± 0.2	98.6 ± 1.3
rPCN_{exp}	—	0.9081	38.6 ± 0.5	—
ParT_{exp}	0.840	0.9121	41.3 ± 0.3	101.2 ± 1.1
$\text{ParticleNet-f.t.}_{\text{exp}}$	0.839	0.9115	40.1 ± 0.2	100.3 ± 1.0
$\textbf{ParT-f.t.}_{\textbf{exp}}$	**0.843**	**0.9151**	$\mathbf{42.4 \pm 0.2}$	$\mathbf{107.9 \pm 0.5}$
PFN_{full}	—	0.9052	37.4 ± 0.7	—
$\text{ABCNet}_{\text{full}}$	0.840	0.9126	42.6 ± 0.4	118.4 ± 1.5
PCT_{full}	0.841	0.9140	43.2 ± 0.7	118.0 ± 2.2
$\text{LorentzNet}_{\text{full}}$	0.844	0.9156	42.4 ± 0.4	110.2 ± 1.3
$\text{ParT}_{\text{full}}$	0.849	0.9203	47.9 ± 0.5	129.5 ± 0.9
$\textbf{ParT-f.t.}_{\textbf{full}}$	**0.852**	**0.9230**	$\mathbf{50.6 \pm 0.2}$	$\mathbf{138.7 \pm 1.3}$

the JetClass dataset. In the “full” scenario, we consider all particle types and further distinguish electrically charged (and neural) hadrons into more types, such as pions, kaons, and protons. We perform the pre-training on JetClass using only kinematic and particle identification inputs under the “exp” scenario. For the fine-tuning, we then carry out experiments in both scenarios. The construction of the input features is described in Table 2. The pre-training and fine-tuning setup is the same as in the top quark tagging benchmark, and the fine-tuning also lasts for 20 epochs. Results are summarized in Table 6. The pre-trained ParT achieves the best performance and improves existing baselines by a large margin in both scenarios.

6. Discussion and Conclusion

Large-scale datasets have always been a catalyst for new breakthroughs in deep learning. In this work, we present JetClass, a new large-scale open dataset to advance deep learning research in particle physics. The dataset consists of 100 M simulated jets, about two orders of magnitude larger than existing public jet datasets, and covers a broad spectrum of 10 classes of jets in total, including several novel types that have not been studied with deep learning so far. While we focus on investigating a classification task, i.e., jet tagging, with this dataset, we highlight that this dataset can serve as the basis for many important deep learning researches in particle physics, e.g., unsupervised or self-supervised training techniques for particle physics (e.g., Dillon et al. (2021)), generative models for high-fidelity fast simulation of particle collisions (e.g., Kansal et al. (2021a)), regression models to predict jet energy and momentum with higher precision (e.g., CMS Collaboration (2020a)), and more. We invite the community to explore and experiment with this dataset and extend the boundary of deep learning and particle physics even further.

With this large dataset, we introduce Particle Transformer (ParT), a new architecture that substantially improves jet tagging performance over previous state-of-the-art. We propose it as a new jet tagging baseline for future research to improve upon. The effectiveness of ParT arises mainly from the augmented self-attention, in which we incorporate physics-inspired pairwise interactions together with the machine-learned dot-product attention. This approach is likely to be effective for other tasks on similar datasets, such as point clouds or many-body systems, especially when prior knowledge is available to describe the interaction or the geometry. On the other hand, one limitation of using the full pairwise interaction matrix is the increase in computational time and memory consumption. Novel approaches for particle (point) embeddings and self-attentions that alleviate the computational cost (e.g., Zhou et al. (2021); Kitaev et al. (2020)) could be an interesting direction for future research.

Acknowledgements

We are grateful to Loukas Gouskos, Qiang Li, and Alexandre De Moor for many helpful discussions. The work of C. Li and S. Qian is supported by National Natural Science Foundation of China under Grants No. 12061141002.

References

Alwall, J., Frederix, R., Frixione, S., Hirschi, V., Maltoni, F., Mattelaer, O., Shao, H. S., Stelzer, T., Torrielli, P., and

Zaro, M. The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations. *JHEP*, 07:079, 2014. doi: 10.1007/JHEP07(2014)079.

ATLAS Collaboration. The ATLAS Experiment at the CERN Large Hadron Collider. *JINST*, 3:S08003, 2008. doi: 10.1088/1748-0221/3/08/S08003.

ATLAS Collaboration. Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. *Phys. Lett. B*, 716:1–29, 2012. doi: 10.1016/j.physletb.2012.08.020.

Bogatskiy, A., Anderson, B., Offermann, J., Roussi, M., Miller, D., and Kondor, R. Lorentz group equivariant neural network for particle physics. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 992–1002. PMLR, 13–18 Jul 2020.

Butter, A. et al. The Machine Learning landscape of top taggers. *SciPost Phys.*, 7:014, 2019. doi: 10.21468/SciPostPhys.7.1.014.

Cacciari, M., Salam, G. P., and Soyez, G. The anti-k_t jet clustering algorithm. *JHEP*, 04:063, 2008. doi: 10.1088/1126-6708/2008/04/063.

Cacciari, M., Salam, G. P., and Soyez, G. FastJet User Manual. *Eur. Phys. J. C*, 72:1896, 2012. doi: 10.1140/epjc/s10052-012-1896-2.

Chen, C., Fragonara, L. Z., and Tsourdos, A. Gapointnet: Graph attention based point neural network for exploiting local feature of point cloud. *Neurocomputing*, 438:122–132, 2021. doi: https://doi.org/10.1016/j.neucom.2021.01.095.

Chen, Y., Huerta, E. A., Duarte, J., Harris, P., Katz, D. S., Neubauer, M. S., Diaz, D., Mokhtar, F., Kansal, R., Park, S. E., Kindratenko, V. V., Zhao, Z., and Rusack, R. A FAIR and AI-ready Higgs boson decay dataset. *Scientific Data*, 9(1):31, 2022. doi: 10.1038/s41597-021-01109-0.

CMS Collaboration. The CMS Experiment at the CERN LHC. *JINST*, 3:S08004, 2008. doi: 10.1088/1748-0221/3/08/S08004.

CMS Collaboration. Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC. *Phys. Lett. B*, 716:30–61, 2012. doi: 10.1016/j.physletb.2012.08.021.

CMS Collaboration. Description and performance of track and primary-vertex reconstruction with the CMS tracker. *JINST*, 9(10):P10009, 2014. doi: 10.1088/1748-0221/9/10/P10009.

CMS Collaboration. A deep neural network for simultaneous estimation of b jet energy and resolution. *Comput. Softw. Big Sci.*, 4(1):10, 2020a. doi: 10.1007/s41781-020-00041-z.

CMS Collaboration. Identification of heavy, energetic, hadronically decaying particles using machine-learning techniques. *JINST*, 15(06):P06005, 2020b. doi: 10.1088/1748-0221/15/06/P06005.

CMS Collaboration. Search for top squark production in fully-hadronic final states in proton-proton collisions at $\sqrt{s} = 13$ TeV. *Phys. Rev. D*, 104(5):052001, 2021. doi: 10.1103/PhysRevD.104.052001.

CMS Collaboration. Search for resonances decaying to three W bosons in proton-proton collisions at $\sqrt{s} = 13$ TeV. 1 2022a. Accepted for publication in Phys. Rev. Lett.

CMS Collaboration. Search for nonresonant pair production of highly energetic Higgs bosons decaying to bottom quarks. 2022b. Submitted to *Phys. Rev. Lett.*

CMS Collaboration. Search for Higgs boson decay to a charm quark-antiquark pair in proton-proton collisions at $\sqrt{s} = 13$ TeV. 2022c. Submitted to *Phys. Rev. Lett.*

de Favereau, J., Delaere, C., Demin, P., Giammanco, A., Lemaître, V., Mertens, A., and Selvaggi, M. DELPHES 3, A modular framework for fast simulation of a generic collider experiment. *JHEP*, 02:057, 2014. doi: 10.1007/JHEP02(2014)057.

de Oliveira, L., Kagan, M., Mackey, L., Nachman, B., and Schwartzman, A. Jet-images — deep learning edition. *JHEP*, 07:069, 2016. doi: 10.1007/JHEP07(2016)069.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. doi: 10.18653/v1/N19-1423.

Dillon, B. M., Kasieczka, G., Olischlager, H., Plehn, T., Sorrenson, P., and Vogel, L. Symmetries, Safety, and Self-Supervision. 8 2021.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Dreyer, F., Soyez, G., and Takacs, A. Quarks and gluons in the Lund plane. *arXiv preprint arXiv:2112.09140*, 12 2021.

Dreyer, F. A. and Qu, H. Jet tagging in the Lund plane with graph networks. *JHEP*, 03:052, 2021. doi: 10.1007/JHEP03(2021)052.

Dreyer, F. A., Salam, G. P., and Soyez, G. The Lund Jet Plane. *JHEP*, 12:064, 2018. doi: 10.1007/JHEP12(2018)064.

Dreyer, F. A., Grabarczyk, R., and Monni, P. F. Leveraging universality of jet taggers through transfer learning. *arXiv preprint arXiv:2203.06210*, 3 2022.

Duarte, J. Sample with jet, track and secondary vertex properties for Hbb tagging ML studies HiggsToBBNTuple_HiggsToBB_QCD_RunII_13TeV_MC, 2019. URL http://opendata.cern.ch/record/12102.

Gong, S., Meng, Q., Zhang, J., Qu, H., Li, C., Qian, S., Du, W., Ma, Z.-M., and Liu, T.-Y. An Efficient Lorentz Equivariant Graph Neural Network for Jet Tagging. *arXiv preprint arXiv:2201.08187*, 1 2022.

Guest, D., Collado, J., Baldi, P., Hsu, S.-C., Urban, G., and Whiteson, D. Jet Flavor Classification in High-Energy Physics with Deep Neural Networks. *Phys. Rev. D*, 94 (11):112002, 2016. doi: 10.1103/PhysRevD.94.112002.

Guo, M.-H., Cai, J.-X., Liu, Z.-N., Mu, T.-J., Martin, R. R., and Hu, S.-M. PCT: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, June 2021. doi: 10.1007/s41095-021-0229-5.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Henrion, I., Brehmer, J., Bruna, J., Cho, K., Cranmer, K., Louppe, G., and Rochette, G. Neural Message Passing for Jet Physics. In *Deep Learning for Physical Sciences Workshop at the 31st Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021. doi: 10.1038/s41586-021-03819-2.

Kansal, R., Duarte, J., Su, H., Orzari, B., Tomei, T., Pierini, M., Touranakou, M., Vlimant, J.-R., and Gunopulos, D. Particle cloud generation with message passing generative adversarial networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 23858–23871, 2021a.

Kansal, R., Duarte, J., Su, H., Orzari, B., Tomei, T., Pierini, M., Touranakou, M., Vlimant, J.-R., and Gunopulos, D. Jetnet, May 2021b. URL https://doi.org/10.5281/zenodo.5502543.

Kasieczka, G., Plehn, T., Thompson, J., and Russel, M. Top Quark Tagging Reference Dataset, March 2019.

Kitaev, N., Kaiser, L., and Levskaya, A. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020.

Kogler, R. et al. Jet Substructure at the Large Hadron Collider: Experimental Review. *Rev. Mod. Phys.*, 91(4): 045003, 2019. doi: 10.1103/RevModPhys.91.045003.

Komiske, P., Metodiev, E., and Thaler, J. Pythia8 Quark and Gluon Jets for Energy Flow, may 2019a.

Komiske, P. T., Metodiev, E. M., and Thaler, J. Energy Flow Networks: Deep Sets for Particle Jets. *JHEP*, 01:121, 2019b. doi: 10.1007/JHEP01(2019)121.

Larkoski, A. J., Moult, I., and Nachman, B. Jet Substructure at the Large Hadron Collider: A Review of Recent Advances in Theory and Machine Learning. *Phys. Rept.*, 841:1–63, 2020. doi: 10.1016/j.physrep.2019.11.001.

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*, 2020.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10012–10022, October 2021.

Louppe, G., Cho, K., Becot, C., and Cranmer, K. QCD-Aware Recursive Neural Networks for Jet Physics. *JHEP*, 01:057, 2019. doi: 10.1007/JHEP01(2019)057.

Mikuni, V. and Canelli, F. ABCNet: An attention-based method for particle tagging. *Eur. Phys. J. Plus*, 135(6): 463, 2020. doi: 10.1140/epjp/s13360-020-00497-3.

Mikuni, V. and Canelli, F. Point cloud transformers applied to collider physics. *Mach. Learn. Sci. Tech.*, 2(3):035027, 2021. doi: 10.1088/2632-2153/ac07f6.

Moreno, E. A., Cerri, O., Duarte, J. M., Newman, H. B., Nguyen, T. Q., Periwal, A. a., Pierini, M., Serikova, A., Spiropulu, M., and Vlimant, J.-R. JEDI-net: a jet identification algorithm based on interaction networks. *Eur. Phys. J. C*, 80(1):58, 2020. doi: 10.1140/epjc/s10052-020-7608-4.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. 2019.

Pierini, M., Duarte, J. M., Tran, N., and Freytsis, M. Hls4ml lhc jet dataset (150 particles), January 2020. URL https://doi.org/10.5281/zenodo.3602260.

Qu, H. and Gouskos, L. ParticleNet: Jet Tagging via Particle Clouds. *Phys. Rev. D*, 101(5):056019, 2020. doi: 10.1103/PhysRevD.101.056019.

Qu, H., Li, C., and Qian, S. JetClass: A large-scale dataset for deep learning in jet physics, June 2022. URL https://doi.org/10.5281/zenodo.6619768.

Radovic, A., Williams, M., Rousseau, D., Kagan, M., Bonacorsi, D., Himmel, A., Aurisano, A., Terao, K., and Wongjirad, T. Machine learning at the energy and intensity frontiers of particle physics. *Nature*, 560(7716): 41–48, 2018. doi: 10.1038/s41586-018-0361-2.

Shimmin, C. Particle Convolution for High Energy Physics. *arXiv preprint arXiv:2107.02908*, 7 2021.

Shleifer, S., Weston, J., and Ott, M. Normformer: Improved transformer pretraining with extra normalization. *arXiv preprint arXiv:2110.09456*, 2021.

Sjöstrand, T., Ask, S., Christiansen, J. R., Corke, R., Desai, N., Ilten, P., Mrenna, S., Prestel, S., Rasmussen, C. O., and Skands, P. Z. An introduction to PYTHIA 8.2. *Comput. Phys. Commun.*, 191:159–177, 2015. doi: 10.1016/j.cpc.2015.01.024.

Touvron, H., Cord, M., Sablayrolles, A., Synnaeve, G., and Jégou, H. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 32–42, October 2021.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.*, 38(5), oct 2019. doi: 10.1145/3326362.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Zhang, M., Lucas, J., Ba, J., and Hinton, G. E. Lookahead optimizer: k steps forward, 1 step back. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, May 2021.