FERMILAB-PUB-23-748-CMS

Efficient and Robust Jet Tagging at the LHC with Knowledge Distillation

Ryan Liu
University of California, Berkeley
Berkeley, CA 94720

Abhijith Gandrakota
Fermi National Accelerator Laboratory
Batavia, IL 60510

Jennifer Ngadiuba
Fermi National Accelerator Laboratory
Batavia, IL 60510

Maria Spiropulu
California Institute of Technology
Pasadena, CA 91125

Jean-Roch Vlimant
California Institute of Technology
Pasadena, CA 91125

Abstract

The challenging environment of real-time data processing systems at the Large Hadron Collider (LHC) strictly limits the computational complexity of algorithms that can be deployed. For deep learning models, this implies that only small models that have low capacity and weak inductive bias are feasible. To address this issue, we utilize knowledge distillation to leverage both the performance of large models and the reduced computational complexity of small ones. In this paper, we present an implementation of knowledge distillation, demonstrating an overall boost in the student models' performance for the task of classifying jets at the LHC. Furthermore, by using a teacher model with a strong inductive bias of Lorentz symmetry, we show that we can induce the same inductive bias in the student model which leads to better robustness against arbitrary Lorentz boost. The code of this work is available in this Github repository

1 Introduction

In the past decades, deep learning has transformed the data analysis workflow at high-energy physics experiments, achieving state-of-the-art performance on many challenging tasks such as jet tagging, charged particle tracking, particle flow algorithm, and neutrino event reconstruction [1–4]. Such breakthroughs mostly follow from novel architectures and large-scale datasets. In particular, architectural innovation involves both general-purposed models adopted from deep-learning research (e.g. convolutional neural networks, transformers, graph neural networks [1, 5, 6]) and custom models that incorporate strong inductive biases [7, 8]. However, these paradigms are not applicable to each stage of the experimental data processing workflow, especially to online applications such as the hardware trigger systems [9, 10] of particle detectors at the LHC that are characterized by strict latency, power, and resource constraints. For such applications, a generic method that can improve deep learning models' performance and robustness without scaling the computation in the production environment becomes crucial.

Since knowledge distillation (KD) was proposed by Hinton in 2015 [11], it has been widely used in natural language processing applications [12, 13]. KD is a technique to transfer the learned knowledge from a large and cumbersome "teacher" model to a more efficient "student" model.

Machine Learning and the Physical Sciences Workshop, NeurIPS 2023.

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

This is done by replacing the ground truth (hard targets) with teacher-predicted probabilities (soft targets). Remarkably, KD can significantly boost the performance of the student model without any modification to the architecture itself. In this paper, we describe an implementation of KD for jet tagging at the LHC, hoping that by demonstrating its efficacy, more powerful deep learning models can be deployed to early-stage event selections.

2 Related Work

2.1 Group Invariant Neural Networks

Inductive bias is a crucial part of designing a neural network. In the context of jet physics, we have two important symmetries that shall be taken into consideration: permutation invariance and Lorentz group invariance. Whereas the constituents of a jet come in no particular order, the prediction should not change upon the exchange of two particles in the input point cloud. At the same time, since the source of the jet can have high momentum and cause the jet to be boosted, the prediction shall not change by an arbitrary Lorentz transformation (referred to as boost) as well.

To incorporate permutation symmetry, it was proven that all permutation invariant functions can be written as follows [14]:

$$f(X) = \rho(\sum_{x \in X} \phi(x)) \tag{1}$$

We can parameterize ρ and ϕ using multi-layer perceptrons (MLPs) to make the network learnable. The resulting architecture is called a *deep set* model. To improve the expressiveness of the group invariant neural network, we can also consider attention-based models such as Transformers or Graph Attention Networks [15, 16].

As for Lorentz symmetry, it is typically harder to incorporate its inductive bias. Existing solutions include restricting the message-passing mechanism of graph neural networks to use only Lorentz scalars such as $\langle x, y \rangle$ or $\|x - y\|^2$ [7] and manipulating representations of $SO^+(1,3)$ by tensor-product and Clebsch–Gordan decomposition [8].

2.2 Knowledge Distillation

One of the common problems of the models described in the previous section is that most of them do not scale well, especially for models that require pair-wise computation such as Transformers and Graph Neural Networks. Therefore, we propose to use KD to leverage the power of these models while not adding to the inference-time cost. KD transfers learned knowledge by replacing the hard target normally used to train the small and fast student model with soft targets from the large and cumbersome teacher model. To be more precise, the KD loss is defined as follows:

$$L_{KD}(q; p, y) = (1 - \lambda)\mathcal{H}(y, q) + \lambda D_{KL}(\tilde{p} \| \tilde{q}) \tag{2}$$

where q is the student's output probabilities, p is the teacher's output probabilities, and y is the ground truth label. Probabilities with $\tilde{\cdot}$ are the distributions softened by temperature T, i.e., $p(x) = \frac{e^{s(x)}}{\sum_{x'} e^{s(x')}}$ means $\tilde{p}(x) = \frac{e^{s(x)}/T}{\sum_{x'} e^{s(x')/T}}$. The first term is the cross entropy loss with the ground truth as the target, and the second term is the Kullback–Leibler divergence with the softened teacher as the target. It was pointed out that the KD loss has two major effects, namely, it can inject the class relationship prior to the student and rescale the gradients of samples according to the teacher's relative confidence [17]. Moreover, recent studies showed that KD is capable of transferring inductive bias [18]. That is, if the teacher has a strong inductive bias, the student can learn from the KD loss to generalize the same way as the teacher does.

3 Experiments

3.1 Top-quark jets Tagging at the LHC

In high energy physics, a *jet* refers to a collimated shower of particles that result from the decay and hadronization of quarks q and gluons g. The goal of *jet tagging* is to identify which mother particle originated the jet such to distinguish interesting and rare signal events from highly frequent background ones. In this paper, we study the effect of KD on a common benchmark dataset for jet tagging, the top tagging dataset [19]. The dataset contains signal top-quark jets and background light quark and gluon jets simulated with Pythia8 [20] and passed through a simulation of a typical LHC particle detector obtained with Delphes [21]. There are 1.2M training events, 400K validation events, and 400K testing events. Each jet is described by its constituents' four momenta, together with a flag indicating if it is a signal event.

3.2 Model Architecture

In this paper, we demonstrate the efficacy of KD on two student models, the deep set model [14] and an MLP model. The inputs are the constituents' momenta in cylindrical coordinates (p_T, η, ϕ, m). Here we appended the particle's true mass m to the input such that the model can distinguish between different particle species. Note that the set of inputs we used here are also available in a typical real-time data processing system thanks to the particle flow algorithms at the Phase 2 CMS detector [22]. For the deep set model, we used a 3-layer MLP with a hidden size of 128 to parameterize both ρ and ϕ, together with batch normalization [23] applied after leaky relu activations [24]. The aggregation follows the design of the Energy Flow Network such that per-particle embeddings are aggregated according to their p_T to enforce IR-safety [25]. For the MLP model, we sort particles according to their p_T. Any jet with more than 128 particles is trimmed to 128 particles while jets with less than 128 particles are zero-padded. This gives $4 \times 128 = 512$ input features in total. We then feed them into an MLP of 3 layers with 512 hidden features each, again equipped with leaky relu and batch normalization.

3.3 Experiment Setup

To understand if KD can improve student performance and transfer inductive biases, we picked the LorentzNet [7] as the teacher model. LorentzNet has a very strong inductive bias of Lorentz-transformations invariance and is one of the best-performing models on the top-tagging dataset. We designed two experiments; (1) we evaluated the efficacy of KD for both the deep set and the MLP models and compared their performance to the ones trained from scratch. We experimented with temperatures of $T \in \{1, 3, 5\}$. (2) To show how KD can transfer inductive bias, we trained the deep set model on an augmented dataset that was boosted by β uniformly sampled from $[0, \beta_{max}]$ along the x-axis and evaluated the models on the boosted jets test set. We also chose $\lambda = 1$ for this experiment to prevent the model from learning from the ground truth labels. We hypothesized that the ability for KD to transfer inductive biases would be further improved when the student is exposed to some corrupted samples that correspond to the teacher's inductive bias. For each model, we trained for 100 epochs with `AdamW` optimizer and a `StepLR` learning rate scheduler.

3.4 Results

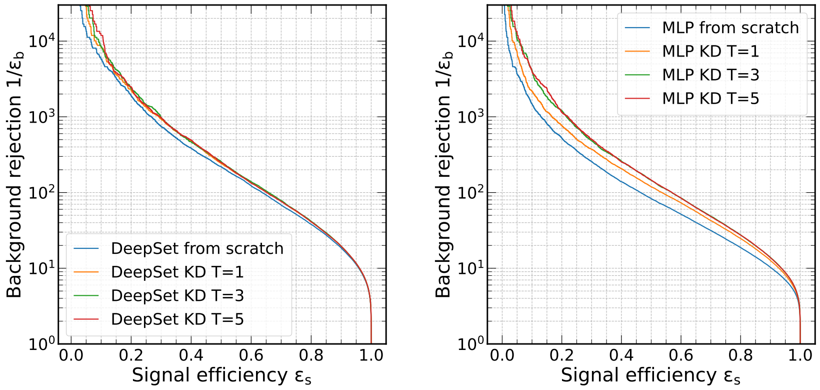
We report the accuracy, the area under curve (AUC), and the background rejection ($\mathrm{R}_{X\%}$) of these models on the test set. The background rejection is defined as $1/\mathrm{FPR}$ when the TPR is fixed to $X\%$. Furthermore, with an eye to deployment to real-time system, we measured the number of floating point operations (FLOPs) with the package `fvcore` [26], which computes a model's FLOPs with torch jit tracing. For the first experiment, the results are reported in Table 1. Both the deep set and MLP models showed performance gains from knowledge distillation. Remarkably, the overall accuracy improvement for the MLP model is 1.5%, together with about a factor of two improvement in background rejection. At the same time, the deep set model showed about a 25% improvement in background rejection. The results demonstrate the effectiveness of knowledge distillation in jet tagging.

For the second experiment, in all three configurations, $\beta_{max} \in \{0, 0.1, 0.3\}$, we can see in Figure 2a that the robustness of the deep set models significantly improved, presumably due to the inductive biases transferred.

Finally, we observed that knowledge distillation can prevent models from overfitting the data (see Figure 2b). This is especially useful when the sample size is small. We hypothesize that this is due to a more complicated learning objective that forces the model to generalize instead of memorize.

Table 1: Comparison between models trained from scratch and knowledge distillation.

	#params	FLOPs	Accuracy	AUC	$\mathrm{Rej}_{30\%}$	$\mathrm{Rej}_{50\%}$
DeepSet from scratch	68.2K	1.67M	0.930	0.9808	747	219
DeepSet KD $T = 1$			**0.932**	0.9818	926	241
DeepSet KD $T = 3$			**0.932**	**0.9819**	**970**	**255**
DeepSet KD $T = 5$			**0.932**	**0.9819**	**970**	248
MLP from scratch	527K	529K	0.904	0.9663	256	82
MLP KD $T = 1$			0.914	0.9726	375	119
MLP KD $T = 3$			0.918	**0.9751**	483	144
MLP KD $T = 5$			**0.919**	0.9750	**503**	**146**
LorentzNet (teacher)	224K	339M	0.942	0.9868	2195	498



(a) ROC curve of Deep sets

(b) ROC curve of MLPs

Figure 1: ROC curves of the two deep set and MLP models on the test dataset. The knowledge distillation models have their curves above the models trained from scratch.

4 Conclusion

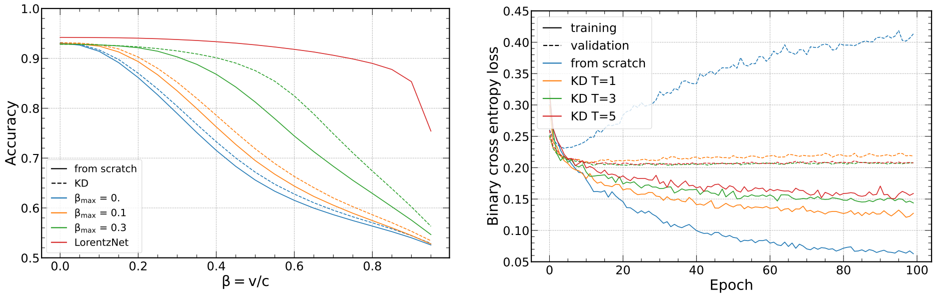
In this paper, we showed that knowledge distillation can improve jet tagging performance and robustness. Without adding any additional operation, we realized a 25% improvement in background rejection. Furthermore, we demonstrated that the teacher's inductive bias can be transferred and help the student generalize better. We hope that this work can serve as a base for future deployment of deep learning models to real-time event selection systems at the LHC, bringing the power of deep learning to the frontier of experimental high-energy physics.

5 Broader Impact Statement

We expect that this work will stimulate the research and further discussions on:

1. **Knowledge distillation between different architectures**. Knowledge distillation enables us to leverage the recent breakthroughs in large models to enhance the small models we have. As shown in this work, knowledge distillation can work for models of different architectures.
2. **Knowledge distillation as a method to improve robustness**. In the Lorentz invariance test, we have seen that knowledge distillation is capable of improving student's robustness. This may suggest that we can train a cumbersome teacher model on an augmented dataset and

[1] The curve here for LorentzNet is different from Figure 3 in [7]. This is because LorentzNet included two auxiliary inputs that represent the beam line. In their paper, the authors also boosted the beams. However, we believe that boosting the beams encodes the information about the boosting so we decided not to follow the convention.



(a) Lorentz invariance test for the deep set model[1].

(b) Training hard target loss

Figure 2: (a) we can see that the robustness with respect to Lorentz boosts improved with KD in all three augmentation configurations. (b) With KD, the validation loss converges instead of increasing as in the case of without KD.

transfer the robustness to the student by knowledge distillation without having the student trained on the augmented dataset, avoiding the accuracy and robustness trade-off.

6 Limitations

There exist a few limitations of this work. Firstly, we have only shown the efficacy of knowledge distillation for a classification problem. Whether it would work for other types of tasks in high-energy physics remains unclear. Secondly, training the teacher model may be very costly. In this work, we used a pre-trained model as the teacher. However, if one would like to train a teacher model from scratch the cost of training may be significant.

References

[1] Huilin Qu, Congqiao Li, and Sitian Qian. Particle transformer for jet tagging. In *International Conference on Machine Learning*, pages 18281–18292. PMLR, 2022.

[2] Sylvain Caillou, Paolo Calafiura, Steven Andrew Farrell, Xiangyang Ju, Daniel Thomas Murnane, Charline Rougier, Jan Stark, and Alexis Vallier. ATLAS ITk Track Reconstruction with a GNN-based pipeline. Technical report, CERN, Geneva, 2022.

[3] Joosep Pata, Javier Duarte, Farouk Mokhtar, Eric Wulff, Jieun Yoo, Jean-Roch Vlimant, Maurizio Pierini, and Maria Girone. Machine learning for particle flow reconstruction at CMS. *Journal of Physics: Conference Series*, 2438(1):012100, feb 2023.

[4] Alexander Shmakov, Alejandro Yankelevich, Jianming Bian, and Pierre Baldi. Interpretable joint event-particle reconstruction for neutrino physics at nova with sparse cnns and transformers, 2023.

[5] CMS collaboration et al. Identification of heavy, energetic, hadronically decaying particles using machine-learning techniques. *Journal of Instrumentation*, 2020.

[6] Jonathan Shlomi, Peter Battaglia, and Jean-Roch Vlimant. Graph neural networks in particle physics. *Machine Learning: Science and Technology*, 2(2):021001, 2020.

[7] 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):1–22, 2022.

[8] Alexander Bogatskiy, Brandon Anderson, Jan Offermann, Marwah Roussi, David Miller, and Risi Kondor. Lorentz group equivariant neural network for particle physics. In *International Conference on Machine Learning*, pages 992–1002. PMLR, 2020.

[9] Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System. Technical report, CERN, Geneva, 2017.

[10] The Phase-2 Upgrade of the CMS Level-1 Trigger. Technical report, CERN, Geneva, 2020. Final version.

[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[12] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*, 2019.

[13] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[14] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander Smola. Deep sets, 2018.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[16] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[17] Jiaxi Tang, Rakesh Shivanna, Zhe Zhao, Dong Lin, Anima Singh, Ed H. Chi, and Sagar Jain. Understanding and improving knowledge distillation, 2021.

[18] Samira Abnar, Mostafa Dehghani, and Willem Zuidema. Transferring inductive biases through knowledge distillation. *arXiv preprint arXiv:2006.00555*, 2020.

[19] Gregor Kasieczka, Tilman Plehn, Jennifer Thompson, and Michael Russel. Top quark tagging reference dataset, March 2019.

[20] 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.

[21] J De Favereau, Christophe Delaere, Pavel Demin, Andrea Giammanco, Vincent Lemaitre, Alexandre Mertens, and Michele Selvaggi. Delphes 3: a modular framework for fast simulation of a generic collider experiment. *Journal of High Energy Physics*, 2014(2):1–26, 2014.

[22] The Phase-2 Upgrade of the CMS L1 Trigger Interim Technical Design Report. Technical report, CERN, Geneva, 2017. This is the CMS Interim TDR devoted to the upgrade of the CMS L1 trigger in view of the HL-LHC running, as approved by the LHCC.

[23] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

[24] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network, 2015.

[25] Patrick T Komiske, Eric M Metodiev, and Jesse Thaler. Energy flow networks: deep sets for particle jets. *Journal of High Energy Physics*, 2019(1):1–46, 2019.

[26] Facebook Research. fvcore. `https://github.com/facebookresearch/fvcore`, 2019.