Accelerating Molecular Graph Neural Networks via Knowledge Distillation

Filip Ekström Kelvinius [*1] **Dimitar Georgiev** [*2] **Artur Petrov Toshev** [*3] **Johannes Gasteiger** [4]

Abstract

Recent advances in graph neural networks (GNNs) have allowed molecular simulations with accuracy on par with conventional gold-standard methods at a fraction of the computational cost. Nonetheless, as the field has been progressing to bigger and more complex architectures, state-of-the-art GNNs have become largely prohibitive for many large-scale applications. In this paper, we, for the first time, explore the utility of knowledge distillation (KD) for accelerating molecular GNNs. To this end, we devise KD strategies that facilitate the distillation of hidden representations in directional and equivariant GNNs and evaluate their performance on the regression task of energy and force prediction. We validate our protocols across different teacher-student configurations and demonstrate that they can boost the predictive accuracy of student models without altering their architecture. Using our KD protocols, we manage to close as much as 60% of the gap in predictive accuracy between models like GemNet-OC and PaiNN with zero additional cost at inference.

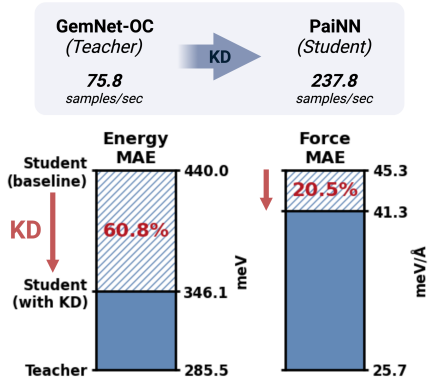

Figure 1. Using knowledge distillation, we manage to substantially boost the predictive accuracy of student models without altering their architecture. This allows us to lessen the tradeoff between speed and performance in molecular GNNs, and run more efficient molecular simulations.

1. Introduction

In the last couple of years, the field of molecular simulations has undergone a rapid paradigm shift with the advent of new, powerful computational tools based on machine learning (ML) (Noé et al., 2020; Westermayr et al., 2021). At the forefront of this transformation have been recent advances in graph neural networks (GNNs), which have brought about architectures that more effectively capture geometric and structural information critical for the accurate representation of molecules and molecular systems (Reiser et al., 2022; Wang et al., 2023). Consequently, a multitude of GNNs have been developed, which now offer predictive performance on par with conventional gold-standard methods like density functional theory (DFT) at a fraction of the computational cost at inference time (Batzner et al., 2022; Gasteiger et al., 2020b; 2021; Musaelian et al., 2023). This has, in turn, significantly accelerated the modeling of molecular properties and the simulation of diverse molecular systems, bolstering new research developments in many scientific disciplines, including material sciences, drug discovery and catalysis.

Nonetheless, this progress - largely coinciding with the development of bigger and more complex models, has naturally come at the expense of increased complexity (Sriram et al., 2022; Zitnick et al., 2022). This has gradually limited the utility of state-of-the-art GNNs for large-scale molecular simulation applications, where inference throughput (i.e., how many samples can be processed for a given time) is critical for making fast continual predictions about the evolution of a system. Hence, addressing the trade-off between accuracy and computational demand remains essential for creating more affordable tools for molecular simulations and expanding the transformational impact of GNN models in the area.

[*]Equal contribution [1]Linköping University [2]Imperial College London [3]Technical University of Munich [4]Google Research. Correspondence to: Filip Ekström Kelvinius <filip.ekstrom@liu.se>, Dimitar Georgiev <d.georgiev21@imperial.ac.uk>, Artur Petrov Toshev <artur.toshev@tum.de>.

Accepted after peer-review at the 1st workshop on Synergy of Scientific and Machine Learning Modeling, SynS & ML ICML, Honolulu, Hawaii, USA. July, 2023.

Motivated by that, in this work, we investigate the potential of knowledge distillation (KD) in enhancing the performance and scalability of state-of-the-art GNNs for molecular simulations. To this end, we devise custom strategies for KD on molecular GNNs, which we call *node-to-node (n2n)*, *edge-to-node (e2n)* and *vector-to-vector (v2v)* knowledge distillation. These overcome common limitations of KD for regression tasks by facilitating the distillation of hidden representations in directional and equivariant GNNs. We evaluate the performance of our KD protocols in augmenting the training process of different student models - without altering their architecture -trained to predict molecular properties like energy and forces. We show that our protocols substantially improve the performance of student models while fully preserving throughput, reducing the performance gap between models like GemNet-OC (Gasteiger et al., 2022) and PaiNN (Schütt et al., 2021) by as much as 60.8% in energy predictions and 20.5% in force predictions (Figure 1).

2. Background

Molecular simulations. In this work, we consider molecular systems at an atomic level, i.e., N atoms represented by their atomic number $\boldsymbol{z} = \{z_1, ..., z_N\} \in \mathbb{Z}^N$ and positions $\boldsymbol{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\} \in \mathbb{R}^{N\times 3}$. Given a system, we want a model that can predict the energy $E \in \mathbb{R}$ of the system, and the forces $\boldsymbol{F} \in \mathbb{R}^{N\times 3}$ acting on each atom. Both these properties are of high interest when simulating molecular systems. The energy of a system is essential for the prediction of its stability, whereas the forces are important for molecular dynamics simulations, where computed forces are combined with the equations of motion to simulate the evolution of the system over time.

GNNs for molecular systems. GNNs are a suitable framework for modeling molecular systems. Each molecular system $(\boldsymbol{X}, \boldsymbol{z})$ can be represented as a mathematical graph, where the set of atoms corresponds to the nodes $\mathcal{V}$, and edges $\mathcal{E}$ created between nodes by connecting the closest neighboring atoms (typically defined by a cutoff radius and/or a maximum number of neighbors). Hence, in the context of molecular simulations, we can create a GNN that operates on atomic graphs $G = (\mathcal{V}, \mathcal{E})$ by propagating information between the atoms and the edges, and makes predictions about the energy and forces of each system in a multi-output manner - i.e., $\hat{E}, \hat{\boldsymbol{F}} = \text{GNN}(\boldsymbol{X}, \boldsymbol{z})$.

The main problem when modeling molecules and molecular properties are the number of underlying symmetries to consider, most importantly rigid transformations of the atoms. For instance, the total energy E of a system is not affected by (i.e., is *invariant* to) rotations or translations of the system. However, the forces $\boldsymbol{F}$ do change as we rotate a system - i.e., they are *equivariant* to rotations. Therefore, to make accurate predictions about molecular systems, it is crucial to devise models that respect these symmetries and other physical constraints. There is now a plethora of diverse molecular GNNs that achieve that, e.g., SchNet (Schütt et al., 2017), DimeNet (Gasteiger et al., 2020b;a), PaiNN (Schütt et al., 2021), GemNet (Gasteiger et al., 2021; 2022), NequIP (Batzner et al., 2022), and SCN (Zitnick et al., 2022).

Knowledge distillation. Knowledge distillation is a technique for compressing and accelerating ML models (Cheng et al., 2018), which has recently demonstrated significant potential in domains like computer vision (Wang & Yoon, 2021) and natural language modeling (Sanh et al., 2019). The main objective of KD is to create more efficient models by means of transferring knowledge (e.g. model parameters and activations) from large, computationally expensive, more accurate models, often referred to as teacher models, to simpler, more efficient models called student models (Gou et al., 2021). Since the seminal work of Hinton et al. (2015), the field has drastically expanded methodologically, with the development of protocols that accommodate the distillation of "deeper" knowledge, more comprehensive transformation and fusion functions, as well as more robust distillation losses (Gou et al., 2021; Hu et al., 2022). Yet, these advances have mostly focused on classification, resulting in methods of limited utility in regression tasks. Moreover, most research in the area has been confined to non-graph data (e.g., images, text, tabular data). Despite recent efforts to extend KD to graph data and GNNs (Tian et al., 2023), these have likewise concentrated on classification tasks involving standard GNN architectures. Hence, the application of KD to state-of-the-art molecular GNN architectures, as well as to real-world regression problems in molecular simulations, is still unexplored.

3. Method

The standard loss function when training molecular GNNs is a loss that combines both the energy and force prediction error as follows:

$$\mathcal{L}_0 = \alpha_{\text{E}}\mathcal{L}_{\text{E}}(\hat{E}, E) + \alpha_{\text{F}}\mathcal{L}_{\text{F}}(\hat{\boldsymbol{F}}, \boldsymbol{F}), \tag{1}$$

where E and $\boldsymbol{F}$ are the ground-truth energy and forces, $\hat{E}$ and $\hat{\boldsymbol{F}}$ are the predictions of the model of interest, and $\mathcal{L}_{\text{E}}$ and $\mathcal{L}_{\text{F}}$ are some loss functions weighted by $\alpha_{\text{E}}, \alpha_{\text{F}} \in \mathbb{R}$.

In KD, we augment this training process by defining an auxiliary knowledge distillation loss term $\mathcal{L}_{\text{KD}}$, which is added to $\mathcal{L}_0$ (with a factor $\lambda \in \mathbb{R}^+$) to derive a new training loss function $\mathcal{L}$ of the form

$$\mathcal{L} = \mathcal{L}_0 + \lambda\mathcal{L}_{\text{KD}}. \tag{2}$$

This was originally proposed in the context of classification by levaraging that the soft label predictions (i.e., the logits af-

ter softmax normalization) of a given (teacher) model carry valuable information that can complement the ground-truth labels in the training process of another (student) model (Hinton et al., 2015). Since then, this has become the standard KD approach - often referred to as vanilla KD in the literature, which is often the foundation of new KD protocols. The main idea of this technique is to employ a KD loss $\mathcal{L}_{\text{KD}}$ that enforces the student to mimic the predictions of the teacher model. This is usually achieved by constructing a loss $\mathcal{L}_{\text{KD}} = \text{KL}(z_s, z_t)$ based on the Kullback–Leibler (KL) divergence between the soft logits of the student z_s and the teacher z_t.

Feature-based KD. Instead of distilling the output only, we focus on feature-based KD (Gou et al., 2021). This is an extension of the vanilla KD, which is concerned with the distillation of knowledge across the intermediate layers of models (Romero et al., 2015). This allows more lightweight models to be trained to mimic representations that can be easier to assimilate compared to the final output directly (Aguilar et al., 2020). In this paper, we perform distillation of intermediate representations by devising a loss on selected hidden features $H_{\text{s}} \in U_{\text{s}}$ and $H_{\text{t}} \in U_{\text{t}}$ in the student and teacher models respectively, which takes the form

$$\mathcal{L}_{\text{KD}} = \mathcal{L}_{\text{feat}}(\mathcal{M}_{\text{s}}(H_{\text{s}}), \mathcal{M}_{\text{t}}(H_{\text{t}})), \tag{3}$$

where $\mathcal{M}_{\text{s}} : U_{\text{s}} \mapsto U$ and $\mathcal{M}_{\text{t}} : U_{\text{t}} \mapsto U$ are transformations that map the hidden features to a common feature space U, and $\mathcal{L}_{\text{feat}} : U \times U \mapsto \mathbb{R}^+$ is some loss of choice. Possible options for the transformations $\mathcal{M}_{\text{s}}, \mathcal{M}_{\text{t}}$ include the identity transformation, linear projections and multilayer perceptron (MLP) projection heads; whereas for the distillation loss $\mathcal{L}_{\text{feat}}$, typical functions are mean squared error (MSE) and mean absolute error (MAE).

Defining feature distillation strategies for molecular GNNs. Unlike standard GNNs that often only consider scalar node features, molecular GNNs can contain diverse features (scalars, vectors and/or equivariant higher-order tensors based on spherical harmonics) organized across nodes and edges within a complex molecular graph. These are continually evolved by model-specific operators to infer molecular properties, such as energy and forces, in a multi-output prediction fashion. Therefore, features often represent different physical, geometric and/or topological information relevant to specific parts of the output. This significantly complicates the design of an effective KD strategy, especially when models differ architecturally.

In this work, we set out to devise KD strategies that are representative and effective across various molecular GNNs. Hence, we consider GNNs that have diverse architectures and performance profiles, namely GemNet-OC (Gasteiger et al., 2022), PaiNN (Schütt et al., 2021), and SchNet (Schütt et al., 2017). Unlike typical student-teacher configurations, these models are characterized by distinct types of features, including scalar node- and edge-features, and equivariant geometrical vectors (see Table 1 for an overview and Appendix A for more information). Here, we leverage these model dissimilarities and devise three distinct KD strategies:

Table 1. Molecular GNNs can have diverse features depending on their architecture. This is an overview of the types of features available in the three models we use in this study.

	SchNet	PaiNN	GemNet-OC
Scalar node features	✓	✓	✓
Scalar edge features			✓
Vectorial node features		✓	
Output blocks			✓

- node-to-node (n2n): As all three models in this study contain scalar node features H_{node}, we can distill knowledge in between these directly by defining a loss $\mathcal{L}_{KD}$, such that

$$\mathcal{L}_{KD} = \mathcal{L}_{\text{feat}}(\mathcal{M}_s(H_{\text{node,s}}), \mathcal{M}_t(H_{\text{node,t}})). \tag{4}$$

Note that this approach only utilizes node features, making it applicable to standard GNNs, too. Previous work has investigated other ways to distill information between node features (e.g. in Joshi et al. (2022); He et al. (2022); Yu et al. (2022)), yet in classification tasks involving simpler, non-molecular GNNs only, which usually share the same architecture. To take advantage of other types of features relevant to molecular GNNs specifically, we further devise two additional protocols below.

- edge-to-node (e2n): The GemNet-OC model relies heavily on edge features, which are a key component in the directional message passing defined in the architecture and can be useful as a KD resource. However, the other models considered here do not have similar edge features to distill to. To accommodate that, we propose a KD strategy where we transfer information from GemNet-OC's edge features $H_{\text{edge},(i,j)}$ by first aggregating them as follows:

$$H_{\text{edge2node},i} = \sum_{j \in \mathcal{N}(i)} H_{\text{edge},(i,j)}, \tag{5}$$

where i is the node index. The resulting vector $H_{\text{edge2node},i}$ is a scalar, node-level feature, and we can, therefore, use it to transfer knowledge to the student node features $H_{\text{node,s}}$ as in Equation (4).

- vector-to-vector (v2v): Similarly, the PaiNN model defines special vectorial features, which differ substantially from the scalar (node and edge) features available in the other models. These are not scalar and invariant to rigid transformations of the atoms, but geometrical vectors that are equivariant with respect to rotations. This poses a new

Table 2. Evaluation of the performance of our KD strategies when distilling information from GemNet-OC into PaiNN. Best results are given in **bold**. All models have been trained on the OC20-2M dataset. Numbers in brackets represent the proportion of the gap between the student and the teacher that has been closed by the respective KD strategy (in %). The validation results presented here are averaged across the four validation datasets available in OC20. The ditto mark (——〃——) indicates no change from previous row. Results for other teachers-student configurations can be found in Appendix B.

	Inference Throughput	OC20 S2EF Validation			
Model	Samples / GPU sec. ↑	Energy MAE meV ↓	Force MAE meV/Å ↓	Force cos ↑	EFwT % ↑
GemNet-OC (*teacher*)	75.8	286	25.7	0.598	1.06
PaiNN (*student*)	237.8	440	45.3	0.376	0.14
Vanilla KD (1)	——〃——	440(0.0%)	43.9(7.1%)	0.378(0.8%)	0.14(0.4%)
Vanilla KD (2)	——〃——	419(13.6%)	114.8(-353%)	0.324(-23.8%)	0.13(-1.3%)
n2n	——〃——	346(**60.8%**)	42.8(12.8%)	0.393(7.4%)	0.26(**13.4%**)
e2n	——〃——	430(6.8%)	41.3(**20.5%**)	0.405(**12.8%**)	0.20(6.1%)
v2v	——〃——	437(1.8%)	42.0(17.1%)	0.397(9.4%)	0.12(-1.6%)

challenge when distilling knowledge from or onto PaiNN. To this end, we define a KD procedure, where we transfer knowledge between (equivariant) vectorial node features and (invariant) scalar edge features. We achieve that by noting that scalar edge features sit on an equivariant 3D grid since they are associated with an edge between two atoms in 3D space. Hence, we can aggregate the edge features $\{H_{\text{edge},(i,j)}\}_{j\in\mathcal{N}}$ corresponding to a given node i into node-level equivariant vectorial features $H_{\text{vec},i}$ by considering the unit vector $\boldsymbol{u}_{ij} = \frac{1}{|\boldsymbol{x}_j - \boldsymbol{x}_i|}(\boldsymbol{x}_j - \boldsymbol{x}_i)$ that defines the direction of the edge (i, j), such that

$$H^{(k)}_{\text{vec},i} = \sum_{j\in\mathcal{N}(i)} \boldsymbol{u}_{i,j} H^{(k)}_{\text{edge},(i,j)}, \tag{6}$$

with the superscript k indicating the channel. This fulfills the condition of equivariance with respect to rotations, as the vector $\boldsymbol{u}$ is equivariant to rotations, and $H^{(k)}_{\text{edge},(i,j)}$ is a scalar, not influencing the direction.

Baseline KD strategies. To validate the performance of our KD strategies, we evaluate their performance against 2 vanilla-based KD approaches suitable for regression tasks.

Vanilla (1): As mentioned above, the main problem with using vanilla KD for regression is the lack of features analogous to logits. One way of adapting vanilla KD for regression is by steering the student to mimic the final output of the teacher directly (Xu et al., 2022):

$$\mathcal{L}_{\text{KD}} = \alpha_{\text{E}}\mathcal{L}_{\text{E}}(\hat{E}_{\text{s}}, \hat{E}_{\text{t}}) + \alpha_{\text{F}}\mathcal{L}_{\text{F}}(\hat{\boldsymbol{F}}_{\text{s}}, \hat{\boldsymbol{F}}_{\text{t}}), \tag{7}$$

where the subscripts $_{\text{s}}$ and $_{\text{t}}$ refer to the predictions of the student and teacher, respectively. Note that, unlike in classification, this approach does not provide much additional information in regression tasks, except for some limited signal about the error distribution of the teacher model (Cheng et al., 2018; Saputra et al., 2019).

Vanilla (2): One way to enhance the teacher signal during training is to consider the fact that many GNNs for molecular simulations make separate atom- and edge-level predictions which are consequently aggregated into a final output. For instance, the total energy E of a system is usually defined as a sum of the predicted contributions from each atom $\hat{E} = \sum_i \hat{E}_i$. Hence, we note that we can extend the aforementioned vanilla KD approach by imposing a loss on these granular predictions instead. Following the energy definition above, the KD loss can be expressed as

$$\mathcal{L}_{\text{KD}} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{\text{E}}(\hat{\text{E}}_{i,\text{s}}, \hat{E}_{i,\text{t}}). \tag{8}$$

These individual energy contributions are not part of the labeled data, but, when injected during training, provide more fine-grained information than the aggregated prediction.

4. Experiments

To evaluate our proposed methods, we perform experiments on the OC20-2M dataset (Chanussot et al., 2021) with the models as implemented in the OC20 codebase[1].

The results of our experiments are summarized in Table 2, which provides information about the performance of the baseline student (PaiNN) and teacher model (GemNet-OC), and how this is influenced by the introduction of different KD protocols. Our analyses demonstrate that KD can effectively boost the predictive performance of the student model for both energy and force predictions without imposing any additional computational constraints at inference time. Using our bespoke KD strategies, we manage to distill

[1] https://github.com/Open-Catalyst-Project/ocp

knowledge from the teacher to the $4\times$ faster student model, while maintaining a substantial part of the predictive accuracy of the teacher. In particular, we note that the *n2n* loss reduces the energy MAE of PaiNN by more than 20%, closing the gap to the teacher GemNet-OC by more than 60%. For force predictions, we observe the biggest improvement with the *e2n* and *v2v* protocols, where with the former we manage to close the performance gap between the two models by more than 20%.

One caveat of knowledge distillation is that it inherently increases the training time of a model. In our offline KD setup, we need to perform additional forward passes through the teacher to extract representations to distill to the student. However, it is important to note that, despite increasing the computational time per training step, we observed that models trained with KD consistently outperformed their baseline counterparts even when compared at the same training time point (Figure 2), despite the latter having been trained for more steps/epoch in total. This means that, all in all, we can use KD to enhance the predictive accuracy in models without necessarily impacting training times.

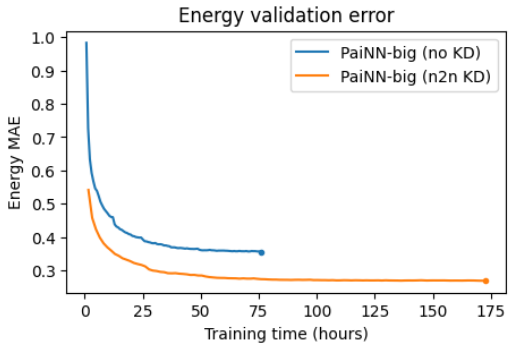

Figure 2. Energy validation error of PaiNN without (*blue*) and with (*orange*) knowledge distillation from GemNet-OC, trained for the same number of steps (1 million). Validation on a random sample of size 30k samples from the in-distribution OC20 validation set.

We also conduct similar analyses with: (1) PaiNN as a teacher model and SchNet as a student model; (2) PaiNN as a teacher model and a smaller version of PaiNN with fewer layers and lower feature dimensions as a student model. The results of these analyses, presented in Appendix B, illustrate that we can successfully employ KD as an effective approach for improving the performance and scalability of molecular GNNs across teacher-student configurations of varying levels of architectural and predictive disparity.

5. Conclusion

In this paper, we investigate the utility of knowledge distillation as a means of distilling larger, more computationally expensive GNNs for molecules into smaller and computationally faster models. We propose three distinct KD strategies and show that they can significantly boost the performance of different GNN models without any modifications to their architecture. Hence, we confirm that KD is a useful technique for addressing the pressing trade-off between predictive accuracy and computational complexity of modern GNNs for molecules. In particular, we demonstrate that KD can allow us to run molecular simulations faster without impairing predictive accuracy. Moreover, we show that our KD strategies are robust and effective in diverse teacher-student configurations. With this work, we aim to highlight the potential of knowledge distillation in enhancing the performance of molecular GNNs and stimulate future research in the area.

Broader impact

The use of molecular GNNs can help to speed up molecular simulations, which find applications in important scientific disciplines, such as material science, drug discovery and catalysis. Note that such applications can be potentially harmful if models are used to simulate and discover systems that are toxic or intended to be used in harmful technology.

Acknowledgements

F. E. K. is financially supported by the Excellence Center at Linköping–Lund in Information Technology (ELLIIT). D.G. is supported by UK Research and Innovation [UKRI Centre for Doctoral Training in AI for Healthcare grant number EP/S023283/1]. This project would not have been possible without the computing resources provided by: the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre (Berzelius resource); the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at Chalmers e-Commons at Chalmers (C3SE) partially funded by the Swedish Research Council through grant agreement no. 2022-06725; as well as the Chair of Aerodynamics and Fluid Mechanics at Technical University of Munich. We are also grateful to the team behind 2022 London Geometry and Machine Learning Summer School (LOGML), where this research project was initially conceived. We would like to specially thank Guocheng Qian and I-Ju Chen for their contribution during the early conceptualizing stages of this project during and in the first weeks following the summer school. We also thank the Open Catalyst team for their open-source codebase, support and discussions. In particular, we would like to thank Muhammed Shuaibi for providing the COLL dataset in LMDB format.

References

Aguilar, G., Ling, Y., Zhang, Y., Yao, B., Fan, X., and Guo, C. Knowledge distillation from internal representations.

In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7350–7357, 2020.

Batzner, S., Musaelian, A., Sun, L., Geiger, M., Mailoa, J. P., Kornbluth, M., Molinari, N., Smidt, T. E., and Kozinsky, B. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13(1):2453, May 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-29939-5.

Chanussot, L., Das, A., Goyal, S., Lavril, T., Shuaibi, M., Riviere, M., Tran, K., Heras-Domingo, J., Ho, C., Hu, W., Palizhati, A., Sriram, A., Wood, B., Yoon, J., Parikh, D., Zitnick, C. L., and Ulissi, Z. The Open Catalyst 2020 (OC20) Dataset and Community Challenges. *ACS Catalysis*, 11(10):6059–6072, May 2021. ISSN 2155-5435, 2155-5435. doi: 10.1021/acscatal.0c04525.

Cheng, Y., Wang, D., Zhou, P., and Zhang, T. Model compression and acceleration for deep neural networks: The principles, progress, and challenges. *IEEE Signal Processing Magazine*, 35(1):126–136, 2018.

Gasteiger, J., Giri, S., Margraf, J. T., and Günnemann, S. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. In *Machine Learning for Molecules Workshop, NeurIPS*, 2020a.

Gasteiger, J., Groß, J., and Günnemann, S. Directional Message Passing for Molecular Graphs. In *International Conference on Learning Representations*, 2020b.

Gasteiger, J., Becker, F., and Günnemann, S. GemNet: Universal Directional Graph Neural Networks for Molecules. In *Advances in Neural Information Processing Systems*, volume 34, pp. 6790–6802. Curran Associates, Inc., 2021.

Gasteiger, J., Shuaibi, M., Sriram, A., Günnemann, S., Ulissi, Z. W., Zitnick, C. L., and Das, A. GemNet-OC: Developing Graph Neural Networks for Large and Diverse Molecular Simulation Datasets. *Transactions on Machine Learning Research*, October 2022. ISSN 2835-8856.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021.

He, H., Wang, J., Zhang, Z., and Wu, F. Compressing deep graph neural networks via adversarial knowledge distillation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 534–544, 2022.

Hinton, G., Vinyals, O., and Dean, J. Distilling the Knowledge in a Neural Network, March 2015.

Hu, C., Li, X., Liu, D., Chen, X., Wang, J., and Liu, X. Teacher-student architecture for knowledge learning: A survey. *arXiv preprint arXiv:2210.17332*, 2022.

Joshi, C. K., Liu, F., Xun, X., Lin, J., and Foo, C. S. On representation knowledge distillation for graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2022. doi: 10.1109/tnnls.2022.3223018. URL https://doi.org/10.1109%2Ftnnls.2022.3223018.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited, 2019.

Musaelian, A., Batzner, S., Johansson, A., Sun, L., Owen, C. J., Kornbluth, M., and Kozinsky, B. Learning local equivariant representations for large-scale atomistic dynamics. *Nature Communications*, 14(1):579, February 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-36329-y.

Noé, F., Tkatchenko, A., Müller, K.-R., and Clementi, C. Machine learning for molecular simulation. *Annual review of physical chemistry*, 71:361–390, 2020.

Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., and Dosovitskiy, A. Do vision transformers see like convolutional neural networks? In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=Gl8FHfMVTZu.

Reiser, P., Neubert, M., Eberhard, A., Torresi, L., Zhou, C., Shao, C., Metni, H., van Hoesel, C., Schopmans, H., Sommer, T., et al. Graph neural networks for materials science and chemistry. *Communications Materials*, 3(1): 93, 2022.

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets, 2015.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Saputra, M. R. U., De Gusmao, P. P., Almalioglu, Y., Markham, A., and Trigoni, N. Distilling knowledge from a deep pose regressor network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 263–272, 2019.

Schütt, K., Kindermans, P.-J., Sauceda Felix, H. E., Chmiela, S., Tkatchenko, A., and Müller, K.-R. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Schütt, K., Unke, O., and Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 9377–9388. PMLR, July 2021.

Schütt, K. T., Kindermans, P.-J., Sauceda, H. E., Chmiela, S., Tkatchenko, A., and Müller, K.-R. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions, 2017.

Sriram, A., Das, A., Wood, B. M., Goyal, S., and Zitnick, C. L. Towards training billion parameter graph neural networks for atomic simulations. *arXiv preprint arXiv:2203.09697*, 2022.

Tian, Y., Pei, S., Zhang, X., Zhang, C., and Chawla, N. V. Knowledge distillation on graphs: A survey, 2023.

Wang, L. and Yoon, K.-J. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Wang, Y., Li, Z., and Farimani, A. B. Graph neural networks for molecules, 2023.

Westermayr, J., Gastegger, M., Schütt, K. T., and Maurer, R. J. Perspective on integrating machine learning into computational chemistry and materials science. *The Journal of Chemical Physics*, 154(23):230903, 2021.

Xu, Q., Chen, Z., Ragab, M., Wang, C., Wu, M., and Li, X. Contrastive adversarial knowledge distillation for deep model compression in time-series regression tasks. *Neurocomputing*, 485:242–251, 2022.

Yu, L., Pei, S., Ding, L., Zhou, J., Li, L., Zhang, C., and Zhang, X. Sail: Self-augmented graph contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8927–8935, 2022.

Zitnick, L., Das, A., Kolluru, A., Lan, J., Shuaibi, M., Sriram, A., Ulissi, Z., and Wood, B. Spherical channels for modeling atomic interactions. *Advances in Neural Information Processing Systems*, 35:8054–8067, 2022.

A. Description of features

- *SchNet* (Schütt et al., 2017): A simple GNN model based on continuous-filter convolutional layers, which only contains scalar node features $\boldsymbol{s} \in \mathbb{R}^d$. These are used to predict the energy, $\hat{E}$. The force is then calculated as the negative gradient of the energy with respect to the atomic positions, i.e., $\hat{\boldsymbol{F}} = -\nabla \hat{E}$.
- *PaiNN* (Schütt et al., 2021): A GNN based on equivariant message passing, which contains scalar node features $\boldsymbol{x} \in \mathbb{R}^{d_1}$ - used for energy prediction; as well as geometric vectorial node features, $\boldsymbol{v} \in \mathbb{R}^{3 \times d_2}$ that are equivariant to rotations and can thus be combined with the scalar features to make direct predictions of the forces (i.e., without computing gradients of the energy).
- *GemNet-OC* (Gasteiger et al., 2022): A GNN model that utilizes directional message passing. It contains scalar node features $\boldsymbol{h} \in \mathbb{R}^{d_\text{h}}$ and scalar edges features $\boldsymbol{m} \in \mathbb{R}^{d_\text{m}}$. After each block of layers, these are processed through an output block, resulting in scalar node features $\boldsymbol{x}_\text{E}^{(i)}$ and edge features $\boldsymbol{x}_\text{F}^{(i)}$, where i is the block number. The output features from each block are aggregated into output features $\boldsymbol{x}_E$ and $\boldsymbol{x}_F$, which are used to compute the energy and forces respectively.

B. Full OC20 results

Here, we present our full results on the OC20-2M S2EF task. We investigated 3 different teacher-student configurations with varying levels of architectural disparity as measured with central kernel alignment (Kornblith et al., 2019; Raghu et al., 2021) (see Figure 3):

- *same* architecture: distilling our default PaiNN model (PaiNN-big) to a smaller version with four instead of six layers, and 256 hidden dimensions instead of 512 (PaiNN-small);
- *similar* architecture: distilling PaiNN-big to SchNet;
- *different* architecture: distilling GemNet-OC to PaiNN-big.

We trained the models as implemented and configured in the OC20 codebase (`https://github.com/Open-Catalyst-Project/ocp`).

Table 3 summarizes the performance of our models without any knowledge distillation, and Table 4 - those with knowledge distillation.

Table 3. Baseline performance of the different GNN models considered in this study without knowledge distillation.

	Inference Throughput	OC20 S2EF Validation			
Model	Samples / GPU sec. ↑	Energy MAE meV ↓	Force MAE meV/Å ↓	Force cos ↑	EFwT % ↑
SchNet	788.2	1308	65.1	0.204	0
PaiNN-small	618.2	489	47.1	0.345	0.085
PaiNN-big	237.8	440	45.3	0.376	0.14
GemNet-OC	75.8	286	25.7	0.598	1.06

Table 4. Model performance with knowledge distillation. The results are averaged across the four OC20 S2EF validation datasets.

		OC20 S2EF Validation			
	Model	Energy MAE meV ↓	Force MAE meV/Å ↓	Force cos ↑	EFwT % ↑
same	Student (PaiNN-small)	489	47.1	0.345	0.085
	Teacher (PaiNN-big)	440	45.3	0.376	0.14
	Vanilla KD (1)	515(−52.4%)	48.5(−81.0%)	0.269(−237%)	0.007(−28%)
	Vanilla KD (2)	476(27.2%)	50.8(−215%)	0.307(−117%)	0.0068(−32.6%)
	n2n	457(**64.8%**)	46.7(**20.5%**)	0.348(**9.3%**)	0.085(**0.5%**)
	v2v	459(60.8%)	47.2(-9.1%)	0.347(6.8%)	0.079(-11.9%)
similar	Student (SchNet)	1308	65.1	0.204	0
	Teacher (PaiNN-big)	440	45.3	0.376	0.14
	Vanilla KD (1)	1214(**10.8%**)	64.6(2.3%)	0.2303(**15.2%**)	0.0025(**1.8%**)
	Vanilla KD (2)	1216(10.5%)	64.6(**2.5%**)	0.229(14.5%)	0(0%)
	n2n	1251(6.6%)	65.2(-0.5%)	0.223(11.1%)	0(0%)
different	Student (PaiNN-big)	440	45.3	0.376	0.14
	Teacher (GemNet-OC)	286	25.7	0.598	1.06
	Vanilla KD (1)	440(0.0%)	43.9(7.1%)	0.378(0.8%)	0.14(0.4%)
	Vanilla KD (2)	419(13.6%)	114.8(-353%)	0.324(-23.8%)	0.13(-1.3%)
	n2n	346(**60.8%**)	42.8(12.8%)	0.393(7.4%)	0.26(**13.4%**)
	e2n	430(6.8%)	41.3(**20.5%**)	0.405(**12.8**%)	0.20(6.1%)
	v2v	437(1.8%)	42.0(17.1%)	0.397(9.4%)	0.12(-1.6%)

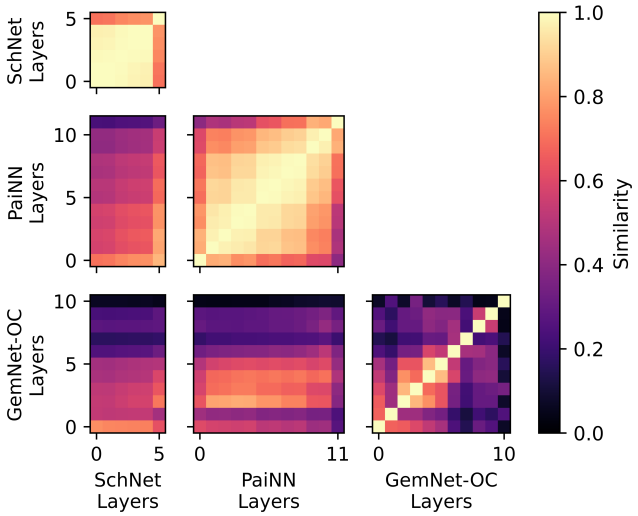

Figure 3. Similarity analysis between the node features of SchNet, PaiNN and GemNet-OC using CKA.