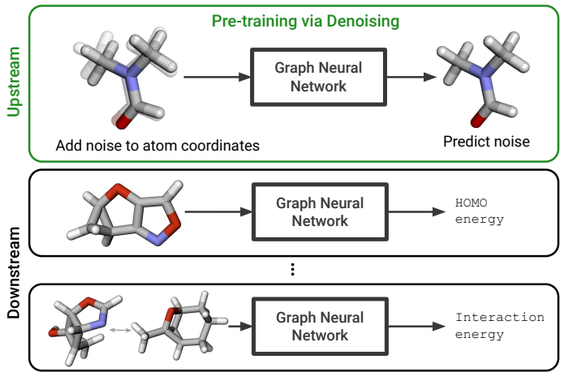
Pre-training via Denoising for Molecular Property Prediction

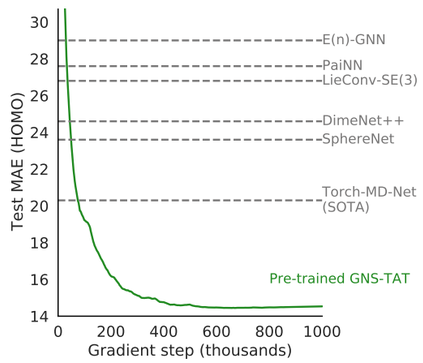
Sheheryar Zaidi*[1†], Michael Schaarschmidt*[2], James Martens[2], Hyunjik Kim[2], Yee Whye Teh[2], Alvaro Sanchez-Gonzalez[2], Peter Battaglia[2], Razvan Pascanu[2], Jonathan Godwin[2]

[1]University of Oxford, [2]DeepMind
[†]Work done during an internship at DeepMind
*Equal contribution

Overview

- We propose a simple and effective technique for pre-training neural networks that take 3D molecular structures as input.
- The pre-training objective is self-supervised and based on *denoising* molecular structures. Denoising structures is shown to be equivalent to learning an approximation of per-atom forces.
- Our experiments demonstrate that pre-training via denoising significantly improves performance on multiple challenging datasets, setting, in particular, a new state-of-the-art on 10 out of 12 targets in QM9.





Pre-training via Denoising

Denoising Molecular Structures

Let $\mathcal{D}_{\text{structures}} = \{S_1, \ldots, S_n\}$ be an (upstream) dataset of equilibrium molecular structures. Each structure $S = \{(a_1, \mathbf{p}_1), \ldots, (a_{|S|}, \mathbf{p}_{|S|})\}$ is a set of atomic nuclei, where a_i is the atomic number and $\mathbf{p}_i \in \mathbb{R}^3$ the spatial coordinate.

Step 1: Perturb molecule coordinates with noise:

$$\tilde{S} = \{(a_1, \tilde{\mathbf{p}}_1), \ldots, (a_{|S|}, \tilde{\mathbf{p}}_{|S|})\},$$
where $\tilde{\mathbf{p}}_i = \mathbf{p}_i + \sigma\boldsymbol{\epsilon}_i$ and $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, I_3)$.

Step 2: Predict the noise per-vertex:

Let GNN_θ denote a neural network backbone returning vertex-level predictions. Minimize the following pre-training objective w.r.t. θ:

$$\mathbb{E}_{p(\tilde{S},S)}\left[\left\|\text{GNN}_\theta(\tilde{S}) - (\boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_{|S|})\right\|^2\right], \quad (1)$$

where $p(\tilde{S}, S)$ is the joint noising distribution.

Fine-tune the model for molecular property prediction by replacing the vertex-level output module with a graph-level one for predicting scalar labels.

Denoising as Learning Forces

Denoising is equivalent to learning an approximation of per-atom forces directly from a dataset of equilibrium structures, explaining why it is useful for representation learning of molecules.

- An equilibrium structure $\mathbf{x} = (\mathbf{p}_1, \ldots, \mathbf{p}_N) \in \mathbb{R}^{3N}$ locally minimizes the energy $E(\mathbf{x})$ and locally maximizes the (Boltzmann) distribution $p_{\text{physical}}(\mathbf{x}) \propto \exp(-E(\mathbf{x}))$.
- The force field $-\nabla_{\mathbf{x}} E(\mathbf{x})$ is equal to the *score function* $\nabla_{\mathbf{x}} \log p_{\text{physical}}(\mathbf{x})$ of p_{physical}.
- Learning the forces corresponds to score-matching:

$$\mathbb{E}_{p_{\text{physical}}(\mathbf{x})}\left[\left\|\text{GNN}_\theta(\mathbf{x}) - \nabla_{\mathbf{x}} \log p_{\text{physical}}(\mathbf{x})\right\|^2\right].$$

- p_{physical} is unknown, but can be approximated with a mixture of Gaussians $q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}_i)$ centered at the local maxima of p_{physical}, *i.e.* equilibrium structures $\mathbf{x}_i \in \mathcal{D}_{\text{structures}}$:

$$p_{\text{physical}}(\tilde{\mathbf{x}}) \approx q_\sigma(\tilde{\mathbf{x}}) := \frac{1}{n}\sum_{i=1}^{n} q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}_i).$$

Conclusion: By the result of Vincent (2011), score-matching with q_σ is equivalent to the denoising objective in eq. (1).

Molecular Property Prediction on QM9

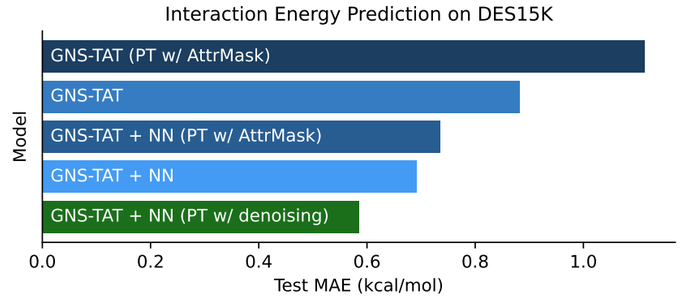
- Pre-train the model on PCQM4Mv2, with over 3 million equilibrium molecular structures.
- Fine-tune the model on QM9, separately for each of the 12 properties.
- We use GNS-TAT as the model, an improved architecture we contribute by applying Tailored Activation Transforms (TAT) to the original GNS.
- Pre-trained GNS-TAT achieves SOTA results on QM9.

Target	Unit	SchNet	E(n)-GNN	DimeNet++	SphereNet	PaiNN	TorchMD-NET	GNS + NN	GNS-TAT + NN	Pre-trained GNS-TAT + NN
μ	D	0.033	0.029	0.030	0.027	0.012	**0.011**	0.025	0.021	0.016
α	a_0^3	0.235	0.071	0.043	0.047	0.045	0.059	0.052	0.047	**0.040**
ϵ_{HOMO}	meV	41.0	29.0	24.6	23.6	27.6	20.3	20.4	17.3	**14.9**
ϵ_{LUMO}	meV	34.0	25.0	19.5	18.9	20.4	18.6	17.5	17.1	**14.7**
$\Delta\epsilon$	meV	63.0	48.0	32.6	32.3	45.7	36.1	28.6	25.7	**22.0**
$\langle R^2 \rangle$	a_0^2	0.07	0.11	0.33	0.29	0.07	**0.033**	0.70	0.65	0.44
ZPVE	meV	1.700	1.550	1.210	1.120	1.280	1.840	1.160	1.080	**1.018**
U_0	meV	14.00	11.00	6.32	6.26	5.85	6.15	7.30	6.39	**5.76**
U	meV	19.00	12.00	6.28	7.33	5.83	6.38	7.57	6.39	**5.76**
H	meV	14.00	12.00	6.53	6.40	5.98	6.16	7.43	6.42	**5.79**
G	meV	14.00	12.00	7.56	8.00	7.35	7.62	8.30	7.41	**6.90**
c_{v}	$\frac{\text{cal}}{\text{mol K}}$	0.033	0.031	0.023	0.022	0.024	0.026	0.025	0.022	**0.020**

Test MAE on QM9. "NN" stands for *Noisy Nodes*, which is denoising applied during fine-tuning as an auxiliary task.

Energy Prediction on DES15K

- DES15K is a small dataset of dimers with non-covalent interactions, labelled with the interaction energies.
- DES15K is generated using CCSD(T), which is more expensive than DFT.
- Pre-train GNS-TAT on PCQM4Mv2, as above with QM9.
- Pre-training via denoising yields best performance. DFT-generated pre-training datasets can even improve downstream performance on CCSD(T)-generated datasets.



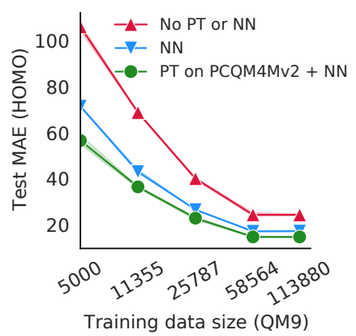
Data and Model Scaling

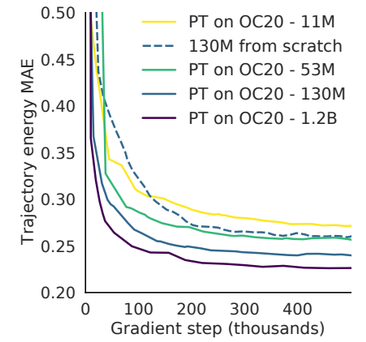
Data Scaling (left plot)

- Using a pre-trained model improves performance across different downstream dataset sizes.
- Pre-training can be especially useful in the small data regime.

Model Scaling (right plot)

- Downstream performance continues to improve as we scale the pre-trained model from 11 million parameters to over a billion parameters on the Open Catalyst 2020 dataset.
- A smaller pre-trained model can outperform a larger model trained from scratch.





More in the paper! See for further experiments, including other datasets and architectures (e.g. TorchMD-NET).

Full paper on arXiv