# Supervised Pretraining for Molecular Force Fields and Properties Prediction

**Xiang Gao, Weihao Gao, Wenzhi Xiao, Zhirui Wang, Chong Wang[*] Liang Xiang**
ByteDance Inc.

{xianggao,weihao.gao,xiaowenzhi}@bytedance.com
{zhirui.wang,xiangliang}@bytedance.com,mr.chongwang@gmail.com

## Abstract

Machine learning approaches have become popular for molecular modeling tasks, including molecular force fields and properties prediction. Traditional supervised learning methods suffer from scarcity of labeled data for particular tasks, motivating the use of large-scale dataset for other relevant tasks. We propose to pretrain neural networks on a dataset of 86 millions of molecules with atom charges and 3D geometries as inputs and molecular energies as labels. Experiments show that, compared to training from scratch, fine-tuning the pretrained model can significantly improve the performance for seven molecular property prediction tasks and two force field tasks. We also demonstrate that the learned representations from the pretrained model contain adequate information about molecular structures, by showing that linear probing of the representations can predict many molecular information including atom types, interatomic distances, class of molecular scaffolds, and existence of molecular fragments. Our results show that supervised pretraining is a promising research direction in molecular modeling

## 1 Introduction

Molecular force fields and properties prediction are important tasks for biotechnology, drug discovery, material and energy science. Machine learning has become a promising approach for these tasks. A weakness of supervised machine learning approaches is that they often require a large amount of labeled data to perform well. However, the molecular datasets are usually small due to the high cost to obtain the labels, which may require wet lab experiments or expensive quantum mechanics simulation.

Pretraining is one approach to alleviate the low-data issue by leveraging relevant large-scale data. The assumption is that the knowledge learned from pretraining can be transferred to the fine-tuning stage on downstream tasks. For computer vision, the most popular pretraining strategy is probably a supervised learning approach, where models are pretrained on image classification tasks using large-scale labeled datasets such as ImageNet [1] and CIFAR [2]. For natural language processing, self-supervised pretraining becomes dominant with the success of the masked language modeling [3] and generative pretraining [4]. Recently, pretraining strategies are explored for molecular modeling [5, 6, 7, 8], and most works employ a self-supervised learning pretraining approach.

Can supervised pretraining help molecular modeling? Besides the data availability issue, another challenge is to choose a proper supervised learning target. Hu et al. [5] experimented with a supervised pretraining strategy, multi-task prediction of more than one thousands biochemical assays. However they observed negative transfer for a few downstream tasks, and suspected that this is

[*] Currently affiliated with Apple Inc., work done at ByteDance Inc.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

due to such supervised pretraining is not "truly-related" to the downstream tasks. We argue that the supervised pretraining tasks should focus on more fundamental physical properties, instead of the biochemical properties that are measured in a complex environment and are relevant to only a narrow range of downstream tasks. We propose to pretrain models to predict molecular energy from molecular structure, with the following considerations.

(i) The supervised pretraining data should be rich. Data for fundamental physical properties, such as molecular energies, are often more abundant. For example, the dataset we use in this work, PubChem PM6 dataset [9], contains energies calculated for 221 millions molecules.

(ii) The supervised pretraining labels should be accurate rather than noisy. Molecular energy can be calculated using established first principle based approach such as density functional theory [10] or semi-empirical method [11]. Therefore the labels can be obtained by the same mechanism for all samples. In contrast, the experimentally measured biochemical assays used in previous work [5] can be noisy due to the difference in experimental conditions or methods across various data sources.

(iii) The supervised pretraining target should be relevant various downstream tasks. Many molecular properties of interest are quantities describing the interaction between molecules and the environment, such as proteins in human body or catalyst in batteries. These processes are governed by the forces acting on the atoms. Molecular energy is relevant as the force acting on an atom is the negative partial gradients of the molecular potential energy with respect to the coordinate of the atom. This close relation between molecular energy and molecular structure encourage the pretrained model to understand and represent molecular structures, which is important to perform well on various downstream tasks [12]. We empirically find that the pretrained model learns to embed various molecular structural information (see Section 4.3).

We further propose a force regularization technique, which minimizes the magnitude of the energy gradient with respect to atom coordinates. This makes the pretrained model more suitable for force field prediction tasks. In contrast, instead of using the exact atom coordinates, most previous pretraining tasks [5, 6, 7, 8] are designed to only use the topological molecular structures. This partially explain why pretraininig approach was not previously used for force field tasks.

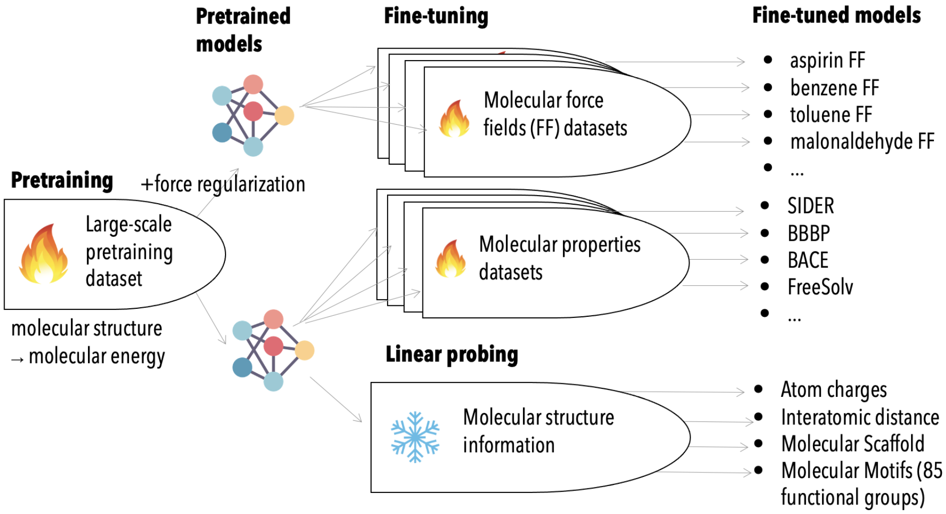

Figure 1: Overview of the proposed pretraining framework.

As illustrated in Figure 1, we find that the proposed pretraining strategy can improve the model performance on various downstream tasks, including molecular force fields and molecular properties prediction such as toxicity and molecular water solubility. Linear probing shows the learned representation can predict the input atom types, interatomic distance, molecular scaffolds, and the existence of 85 functional groups.

Our contribution is three-folds.

1. We propose a supervised pretraining strategy which can improve molecular modeling performance. The pretraining task is to predict molecular energy, a fundamental physical quantity relevant to various downstream tasks and available in large-scale dataset.

2. We show that, without manual design or constraints, the pretrained model learn to embed the molecular structural information. This is necessary to a wide range of downstream molecular properties prediction tasks.

3. We show that the proposed pretraining approach helps force fields models generalize better to unseen molecules compared to training from scratch.

## 2 Background

**Molecular force field.** This task of predicting the force vector acting on each atom in a molecule. The force depends on the positions of the atoms (conformation). Chmiela et al. [13] proposed a MD17 dataset containing the energy-conservative force fields data for a few species. An gradient-domain machine learning (GDML) approach is proposed in [13]. More recently, Klicpera et al. [14] proposed a GemNet model and obtained better results compared to a few previous works. Pretraining has not been employed in existing works for this task.

**Molecular properties** There are enormous molecular properties of interest. Moleculenet [15] collected various datasets of molecular properties on quantum mechanics, physical chemistry [16, 17, 18], biophysics [19], and physiology [20, 21, 22]. Existing works mostly represent the molecule as graphs, with atoms as node and chemical bonds as edges, and then model the molecules with graph neural networks [23, 24, 25, 26] or Transformer[27] based architectures [28, 7].

**Molecular substructures** Predicting the molecular properties from their structures have been explored extensively [12]. Classic methods often investigate molecule structures from two levels, the scaffold [29], and fragments (or motifs), and it is believed that the molecular properties are significantly affected by these substructures [12]. The scaffold has been used to group the molecules and split the train/validation/test sets to mimic a more realistic and challenging setting [15]. The prediction of the existence of fragments have been used in recent deep learning based methods [7] in a un-supervised approach.

**Molecular pretraining** The pretraining-then-fine-tuning paradigm has been used in molecular modeling, with a focus on self-supervised learning. Hu et al. [5] experimented with both node-level and graph-level pre-training. The node-level tasks include prediction of the context of a node, and the prediction of masked node attributes. For graph-level tasks, they experimented with supervised graph-level property prediction, and graph structural similarity prediction. Rong et al. [7] demonstrated two self-supervised learning pretraining tasks: contextual property prediction and graph-level motif prediction. Li et al. [6] proposed a self-supervised pretraining task named pairwise subgraph discrimination, which compares two subgraphs and discriminate whether they come from the same source. Liu et al. [8] aligns the latent space learned for molecular graphs and 3D Geometry in attempt to learn knowledge from both input formats.

## 3 Method

We consider two families of downstream tasks: molecular force field prediction, and molecular properties prediction.

For the former task, we consider models that take in molecular structure, usually the charges $Z$ (e.g., hydrogen, carbon or oxygen) and 3D coordinates of the atoms $R$, as input, and output the forces $F$ acting on each atom. The force for an atom is a 3D vector, usually sharing the same frame of reference as the atom coordinates.

For the latter task, we consider models that take in molecular structure, and output $y$, which is a scalar value (for regression tasks) or a probability distribution (for classification tasks).

Our approach follows the popular pretraining-and-finetuning paradigm.

### 3.1 Pretraining

We use the PubChem PM6 dataset [9] for pretraining. Energy for 86 millions of optimized molecular 3D geometry at neutral states are provided in this dataset. One limitation of our approach is that the performance is affected by the choice of pretraining dataset. The PM6 dataset contain relatively small molecules which makes the pretrained model not suitable for large molecules such as protein or catalysts.

#### 3.1.1 Molecular force fields

We train the model to predict the energy $E$ of the optimized 3D molecular geometry $R$. We define a loss term $\mathcal{L}_E$ based on a distance measure, $d$, between the predicted energy $\hat{E}$ and the energy label value $E$.

$$\mathcal{L}_E = d\left(E, \hat{E}(Z, R)\right),$$

We use mean absolute error as $d$ for $\mathcal{L}_E$.

The PubChem PM6 dataset is built for optimized molecules geometry. The molecular energy is minimized with respect to the atom coordinates. Therefore the energy gradient with respect to the atom coordinates should be close to zero. This motivates us to include a regularization loss term

$$\mathcal{L}_{\partial\hat{E}/\partial R} = ||\frac{\partial\hat{E}(Z, R)}{\partial R}||.$$

As this gradient is actually negative potential forces acting on the atoms, we refer $\mathcal{L}_{\partial\hat{E}/\partial R}$ as force regularization.

The pretraining loss for force fields (FF) task is a linear combination of two loss items.

$$\mathcal{L}_{\text{pretrain, FF}} = (1-\alpha)\mathcal{L}_E + \alpha\mathcal{L}_{\partial\hat{E}/\partial R},$$

where $\alpha$ is a hyperparameter.

#### 3.1.2 Molecular properties

For the molecular properties prediction tasks, many downstream datasets often provide molecular structures in SMILES formats without exact atom coordinates. We instead use an estimated 3D geometry $R_{\text{noisy}}$ obtained from SMILES using tools such as RDKit [30].

$$\mathcal{L}_{E_{\text{noisy}}} = d\left(E, \hat{E}(Z, R_{\text{noisy}})\right),$$

In this case the atom coordinates are approximated, so we do not apply the force regularization term. The pretraining loss is

$$\mathcal{L}_{\text{pretrain, properties}} = \mathcal{L}_{E_{\text{noisy}}}.$$

### 3.2 Fine-tuning

During fine-tuning, we use the parameters of the pretrained model to initialize the model parameters, except these for the final output layers. For the final output layers parameters, we use random initialization.

For molecular properties prediction tasks, the training loss measure the distance $d$ between the label output $y$ and predicted output $y$.

$$\mathcal{L}_{\text{finetune, property}} = d(y, \hat{y}),$$

where the choice of function $d$ depends on the tasks. We use mean absolute error for regression tasks, and cross entropy loss for classification tasks.

For molecular force field prediction tasks, we use a linear combination of the energy and force loss.

$$\mathcal{L}_{\text{finetune, FF}} = (1-\gamma)\mathcal{L}_E + \gamma\mathcal{L}_F,$$

where $\gamma$ is a hyperparameter.

# 4 Experiments and discussion

## 4.1 Molecular force fields

For molecular force fields prediction tasks, the existing works [14, 31, 32] generally employ an in-domain setting, where the molecule type in training set is the same as the test set.

Besides this conventional setting, we consider a more challenging, out-of-domain setting. The models are tested on the force field of a unseen molecule. This is motivated by distinct characteristics of the pretraining and fine-tuning datasets. The PubChem PM6 dataset used in pretraining contains a large number of molecules but each only has one conformation. In contrast, the molecular force field dataset contains many conformations for a limited number of molecules. We wonder if we can combine the knowledge related to different molecules in pretraining dataset and the knowledge of change in conformation in force field dataset. If so, the pretraining should help the model to generalize better on unseen molecules.

We use GemNet-T [14] as the backbone model and use the same set of hyperparameters as [14]. The force is computed as the negative gradient of the predicted energy with respect to atom coordinates. The experiments are conducted with a Nvidia V100 GPU.

### 4.1.1 In-domain test

In this setting, the training data and test data contains different set of conformational geometries and corresponding force fields for the same molecule. We train models for each molecule in the MD17@CCSD dataset [33]. We follow [14] for the train/validation/test split, using 950 samples for training, 50 samples for validation, and 500 samples for test. As shown in Table 1, the models finetuned from the pretrained model generally perform better than the models trained from scratch.

Table 1: The mean absolute error (MAE) in meV/Å for MD17@CCSD molecular force fields prediction task in a in-domain setting.

| | sGDML<br>[31] | NequIP<br>[32] | GemNet-T<br>[14] | GemNet-T,<br>finetuned |
|---|---|---|---|---|
| Aspirin | 33.0 | 14.7 | 10.3 | **9.3** |
| Benzene | 1.7 | 0.8 | 0.8 | **0.7** |
| Ethanol | 15.2 | 9.4 | **3.3** | **3.3** |
| Malonaldehyde | 16.0 | 16.0 | 5.9 | **5.7** |
| Toluene | 9.1 | 4.4 | 2.8 | **2.6** |

### 4.1.2 Out-of-domain test

In this setting, we employ the MD17 DFT dataset [13], which contains 10 kinds of molecules. We construct the training and testing data in a “train-9-test-1" way. For each kind of molecule to be tested, we use the samples from the other 9 molecules as the training set. The test set contains 5k randomly chosen samples, and the training set is built by randomly 50k from each training molecule and mixing the combined 450 samples.

For comparison, we define a naïve constant baseline which always predicts zero force. The reported test error for this baseline is equivalent to the element-wise mean absolute value of the force vectors. As shown in Table 2, the test error is generally much higher than the in-domain test. This is expected as the test molecules are not included in the training set. Benzene and toluene show a relative low loss, probably because benzene ring, their main structural component, has appeared in several molecules (aspirin, azobenzene, salicylic and naphthalene) in the training set. This implies that the existence of similar substructure in training data significantly help the test performance.

The models finetuned from the pretrained model perform significantly better than the models trained from scratch. If the pretraining is conducted without force regularization ($\alpha = 1$), the improvement become less significant. Force regularization encourage the pretrained model not only to learn the energy for a given geometry, but also the energy gradient at this geometry. This makes pretraining more relevant to the force field prediction tasks.

Table 2: The MAE in meV/Å for MD17@DFT for molecular force fields prediction task in a out-of-domain setting. We train the model on 9 molecules and test on 1 unseen molecule.

| Tested unseen molecule | Aspirin | Azobenzene | Benzene | Ethanol | Malonaldehyde |
|---|---|---|---|---|---|
| Const. baseline | 0.899 | 0.910 | 0.626 | 0.841 | 0.907 |
| From scratch | 0.371 | 0.266 | 0.015 | 0.309 | 0.695 |
| w/o. force regularization | 0.248 | 0.220 | 0.013 | 0.318 | **0.557** |
| w. force regularization | **0.205** | **0.215** | **0.013** | **0.260** | 0.602 |
| Tested unseen molecule | Naphthalene | Paracetamol | Salicylic | Toluene | Uracil |
| Const. baseline | 0.884 | 0.885 | 0.891 | 0.866 | 0.921 |
| From scratch | 0.118 | 0.174 | 0.231 | 0.092 | 0.278 |
| From pretrained w/o F | 0.112 | 0.174 | 0.227 | 0.075 | 0.278 |
| From pretrained w. F | **0.104** | **0.165** | **0.191** | **0.065** | **0.241** |

## 4.2 Molecular properties

Following previous works [6, 7, 15], we use scaffold splitting a ratio for train/validation/test as 8:1:1. This splitting method make the molecules in train set do not share molecular scaffold [29] with the molecules in validation or test set. The tested molecules are therefore not "similar" to the molecules in training dataset, and this splitting method is believed to offer a more challenging yet realistic way compared to random splitting. For each dataset, we run three times to obtain three different scaffold splits. The average value and standard deviation of the test metrics are reported.

We use EGNN [34] as the backbone model. If a dataset include more than one tasks (e.g., SIDER contains 27 tasks), we use a multi-task approach, using a single model to predict all tasks simultaneously, similar to [7]. The checkpoint cooresponding to the lowest validation loss is evaluated on the test set. We observe that pretraining makes training on downstream task reaches the lowest validation loss much faster than training from scratch, reducing the necessary training epochs from about 300 to 30.

As shown in Table 3, the models finetuned from the pretrained model generally perform much better than the models trained from scratch. We do not observe negative transfer as Hu et.al. [5] did with their multi-task supervised pretraining strategies. Compared to self-supervised pretraining methods (N-GRAM [35], Hu et.al. [5], GROVER [7], and MPG [6]), our supervised pretraining method achieve similar or better results.

Table 3: The performance comparison for molecular properties prediction tasks. The numbers in brackets are the standard deviation.

| | Classification (AUC-ROC) | | | | Regression (RMSE) | | |
|---|---|---|---|---|---|---|---|
| Dataset | BBBP | SIDER | ClinTox | BACE | FreeSolv | ESOL | Lipo |
| # Molecules | 2039 | 1427 | 1478 | 1513 | 642 | 1128 | 4200 |
| TF_Robust [36] | 0.860$_{(0.087)}$ | 0.607$_{(0.033)}$ | 0.765$_{(0.085)}$ | 0.824$_{(0.022)}$ | 4.122$_{(0.085)}$ | 1.722$_{(0.038)}$ | 0.909$_{(0.060)}$ |
| GraphConv [23] | 0.877$_{(0.036)}$ | 0.593$_{(0.035)}$ | 0.845$_{(0.051)}$ | 0.854$_{(0.011)}$ | 2.900$_{(0.135)}$ | 1.068$_{(0.050)}$ | 0.712$_{(0.049)}$ |
| Weave [37] | 0.837$_{(0.065)}$ | 0.543$_{(0.034)}$ | 0.823$_{(0.023)}$ | 0.791$_{(0.008)}$ | 2.398$_{(0.250)}$ | 1.158$_{(0.055)}$ | 0.813$_{(0.042)}$ |
| SchNet [38] | 0.847$_{(0.024)}$ | 0.545$_{(0.038)}$ | 0.717$_{(0.042)}$ | 0.750$_{(0.033)}$ | 3.215$_{(0.755)}$ | 1.045$_{(0.064)}$ | 0.909$_{(0.098)}$ |
| MPNN [24] | 0.913$_{(0.041)}$ | 0.595$_{(0.030)}$ | 0.879$_{(0.054)}$ | 0.815$_{(0.044)}$ | 2.185$_{(0.952)}$ | 1.167$_{(0.430)}$ | 0.672$_{(0.051)}$ |
| DMPNN [25] | 0.919$_{(0.030)}$ | 0.632$_{(0.023)}$ | 0.897$_{(0.040)}$ | 0.852$_{(0.053)}$ | 2.177$_{(0.914)}$ | 0.980$_{(0.258)}$ | 0.653$_{(0.046)}$ |
| MGCN [26] | 0.850$_{(0.064)}$ | 0.552$_{(0.018)}$ | 0.634$_{(0.042)}$ | 0.734$_{(0.030)}$ | 3.349$_{(0.097)}$ | 1.266$_{(0.147)}$ | 1.113$_{(0.041)}$ |
| AttentiveFP [39] | 0.908$_{(0.050)}$ | 0.605$_{(0.060)}$ | 0.933$_{(0.020)}$ | 0.863$_{(0.015)}$ | 2.030$_{(0.420)}$ | 0.853$_{(0.060)}$ | 0.650$_{(0.030)}$ |
| N-GRAM [35] | 0.912$_{(0.013)}$ | 0.632$_{(0.005)}$ | 0.855$_{(0.037)}$ | 0.876$_{(0.035)}$ | 2.512$_{(0.190)}$ | 1.100$_{(0.160)}$ | 0.876$_{(0.033)}$ |
| Hu. et.al[5] | 0.915$_{(0.040)}$ | 0.614$_{(0.006)}$ | 0.762$_{(0.058)}$ | 0.851$_{(0.027)}$ | - | - | - |
| GROVER[7] | **0.940**$_{(0.019)}$ | 0.658$_{(0.023)}$ | 0.944$_{(0.021)}$ | 0.894$_{(0.028)}$ | 1.544$_{(0.397)}$ | 0.831$_{(0.120)}$ | 0.560$_{(0.035)}$ |
| MPG [6] | 0.922$_{(0.012)}$ | **0.661**$_{(0.007)}$ | **0.963**$_{(0.028)}$ | **0.920**$_{(0.013)}$ | 1.269$_{(0.192)}$ | 0.802$_{(0.023)}$ | 0.576$_{(0.029)}$ |
| EGNN [34] | 0.896$_{(0.030)}$ | 0.646$_{(0.015)}$ | 0.779$_{(0.089)}$ | 0.868$_{(0.046)}$ | 1.421$_{(0.239)}$ | 0.651$_{(0.032)}$ | 0.738$_{(0.072)}$ |
| EGNN, finetuned | **0.948**$_{(0.010)}$ | **0.665**$_{(0.011)}$ | **0.974**$_{(0.017)}$ | **0.934**$_{(0.016)}$ | **0.844**$_{(0.025)}$ | **0.511**$_{(0.013)}$ | **0.510**$_{(0.017)}$ |

## 4.3 What have the pretrained models learned?

The experiments above show that pretraining on energy prediction tasks improve the model performance on both force fields and molecular properties prediction. The improvement on the force fields seems not surprising as the energy is closely related to force, making the pretraining task relevant to the downstream tasks. However, the relation between energy and the tested molecular properties is not so obvious. This makes us wondering what knowledge the pretrained models have learned. Both the molecular energy and the molecular properties depend on the molecular structure [12, 40]. This motivates us to test whether the pretrained model learns to embed the structural information, which can be then used in downstream tasks for molecular properties prediction.

Following [41], we employ linear probes to analyze representation learned by the pretrained EGNN model. Each layer in EGNN model output a node representation $h \in \mathbb{R}^D$ for each atom, where $D$ is the hidden dimension. The node representation can be summed up as the graph representation. For linear probing, the pretrained model is frozen. We use a trainable linear layer to map the learned representation to the output space.

We consider the following two categories of diagnostic tasks to test whether the pretrained model learns to embed the molecular structural information.

### 4.3.1 Reconstructing the input

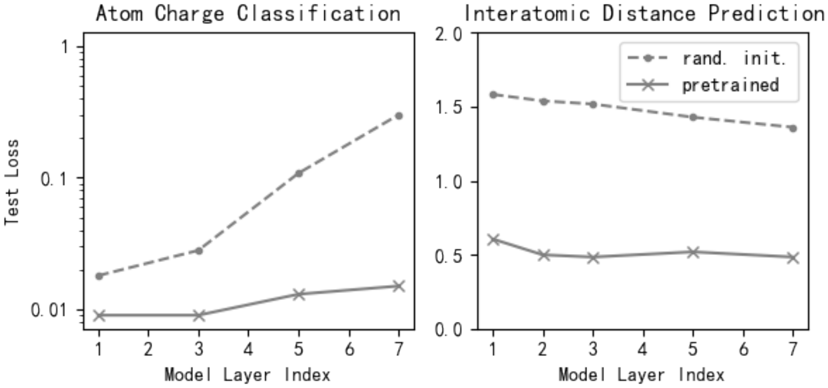

Figure 2: Linear probing on input reconstruction tasks

We firstly test whether the representation produced by the pretrained model can be used to reconstruct the input of EGNN, the atom types and interatomic distance.

**Atom charge classification.** We start with a simple test, whether the node representation contains the information about atom charges (e.g. hydrogen, carbon or oxygen). For a molecule with $n$ atoms, the node representation for the $i$-th atom, $h_i$, is sent to a linear probe layer to output $p_i$ the probability distribution of atom charges.

**Interatomic distance prediction.** This task tests whether the pretrained model learns to embed the 3D geometries. $h_i$ is input to a linear probe layer to output $\hat{x}$, the estimated 3D coordinates.

We compare the test error of these three diagnostic tasks with EGNN models of random parameters. As illustrated in Figure 2, the probing performance of the pretrained model is significantly better than model of random parameters. This indicates that the pretrained model learns to embed structure information, although it is only trained to predict molecular energy.

For atom type classification task, the test loss using the representation from the first layer is smaller than that of the last layer, as shown in Figure 2. This is expected as in EGNN, the atom type information is only directly provided to the first layer. Model of random parameters quickly lose this information as indicated by the significantly higher loss. In contrast, the pretrain model learned to pass this information to the last layer with smaller loss. For interatomic distance prediction task, as the atom coordinates information is sent to each EGNN layer, the test loss do not change significantly across different layers.

The results above show that, although only trained to predict molecular energy, the pretrained model learn to embed molecular structural information in the learned representation. As structural information is important to various downstream tasks, pretraining improve the performance compared to training from scratch.

### 4.3.2 Identifying substructure

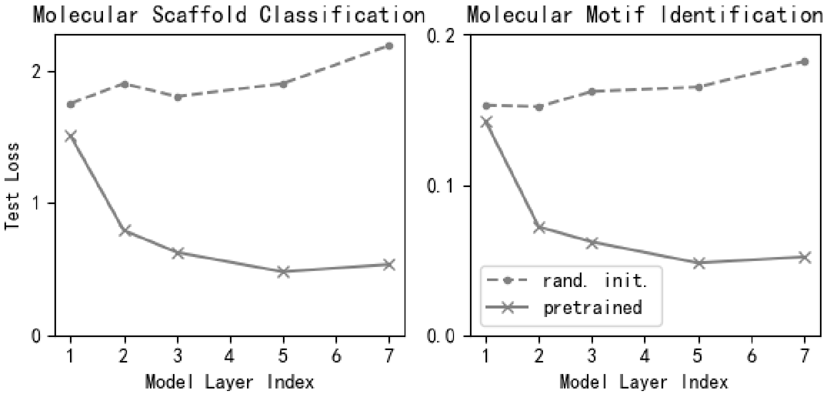

Figure 3: Linear probing on molecular substructure prediction tasks

We then test if the pretrained model learns to embed the information necessary to identify the molecular substructures.

**Molecular scaffold classification.** We are interested in higher level molecular structure, scaffold. From the training data, we choose the top 100 frequent scaffold. With trainable linear layers, parametrized by $W_s^T h \in \mathbb{R}^{D\times D}$ and $W_s^T p \in \mathbb{R}^{100\times D}$, The learned representation is used to predict the scaffold class probability distribution $p_s$. Given labels $y_{\text{scaffold}}$, the probe layers are trained with negative log likelihodd loss.

**Molecular motif identification.** Similar to [7], we consider 85 functional groups [2], such as aliphatic carboxylic acids and H-pyrrole nitrogens, as the molecular motifs. We test if the pretrained model embed the information to identify whether each of these fragment appear in the molecule. We formulate this as a multi-task binary classification problem. With trainable linear layers, parametrized by $W_f^T h \in \mathbb{R}^{D\times D}$, $W_{\text{negative}}^T \in \mathbb{R}^{85\times D}$, and $W_{\text{positive}}^T \in \mathbb{R}^{85\times D}$, the learned representation is used to predict the binary classification probability $p_j$ for the $j$-th fragment. Given labels $y_{\text{motif}}$, the probe layers are trained with negative log likelihodd loss.

Similar to the input reconstruction tasks, the probing performance of the pretrained model is significantly better than model of random parameters, as shown in Figure 2. For molecular scaffold classification task, using the representation from the last EGNN layer of the pretrained model results in a much smaller test error compared to that of the first layer, as shown in Figure 2. Scaffold class is not a input of the EGNN model, and have to be predicted based on the molecular global structure. This implies that the pretrained model learns to gradually obtain the global representation of the molecules at later layers. Similar trends are observed for molecular fragment identification task. As the information pass along to deeper layers, the representation embed more global information and is easier for prediction of molecular subtructure tasks.

## 5 Conclusion

We proposed and demonstrated a supervised learning pretraining strategy for molecular force fields and properties prediction. We find that the pretrained model not only learns to perform the energy prediction task, but also embed the molecular structure information. Experiments show that, compared to training from scratch, fine-tuning the pretrained model can significantly improve the performance. The demonstration covers seven molecular properties datasets and two molecular force field datasets, including a new zero-shot test. New state-of-the-art performance are observed for a few tasks.

[2] Full list: http://rdkit.org/docs/source/rdkit.Chem.Fragments.html

# References

[1] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[2] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[5] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265*, 2019.

[6] Pengyong Li, Jun Wang, Yixuan Qiao, Hao Chen, Yihuan Yu, Xiaojun Yao, Peng Gao, Guotong Xie, and Sen Song. Learn molecular representations from large-scale unlabeled molecules for drug discovery. *arXiv preprint arXiv:2012.11175*, 2020.

[7] Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang. Self-supervised graph transformer on large-scale molecular data. *Advances in Neural Information Processing Systems*, 33:12559–12571, 2020.

[8] Shengchao Liu, Hanchen Wang, Weiyang Liu, Joan Lasenby, Hongyu Guo, and Jian Tang. Pre-training molecular graph representation with 3d geometry. *arXiv preprint arXiv:2110.07728*, 2021.

[9] Maho Nakata, Tomomi Shimazaki, Masatomo Hashimoto, and Toshiyuki Maeda. Pubchemqc pm6: data sets of 221 million molecules with optimized molecular geometries and electronic properties. *Journal of Chemical Information and Modeling*, 60(12):5891–5899, 2020.

[10] Robert G Parr. Density functional theory. *Annual Review of Physical Chemistry*, 34(1):631–656, 1983.

[11] James JP Stewart. Optimization of parameters for semiempirical methods v: Modification of nddo approximations and application to 70 elements. *Journal of Molecular modeling*, 13(12):1173–1213, 2007.

[12] Chanin Nantasenamat, Chartchalerm Isarankura-Na-Ayudhya, Thanakorn Naenna, and Virapong Prachayasittikul. A practical overview of quantitative structure-activity relationship. 2009.

[13] Stefan Chmiela, Alexandre Tkatchenko, Huziel E Sauceda, Igor Poltavsky, Kristof T Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science advances*, 3(5):e1603015, 2017.

[14] Johannes Klicpera, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *arXiv preprint arXiv:2106.08903*, 2021.

[15] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

[16] David L Mobley and J Peter Guthrie. Freesolv: a database of experimental and calculated hydration free energies, with input files. *Journal of computer-aided molecular design*, 28(7):711–720, 2014.

[17] John S Delaney. Esol: estimating aqueous solubility directly from molecular structure. *Journal of chemical information and computer sciences*, 44(3):1000–1005, 2004.

[18] David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, et al. Chembl: towards direct deposition of bioassay data. *Nucleic acids research*, 47(D1):D930–D940, 2019.

[19] Govindan Subramanian, Bharath Ramsundar, Vijay Pande, and Rajiah Aldrin Denny. Computational modeling of $\beta$-secretase 1 (bace-1) inhibitors using ligand based approaches. *Journal of chemical information and modeling*, 56(10):1936–1949, 2016.

[20] Ines Filipa Martins, Ana L Teixeira, Luis Pinheiro, and Andre O Falcao. A bayesian approach to in silico blood-brain barrier penetration modeling. *Journal of chemical information and modeling*, 52(6):1686–1697, 2012.

[21] Michael Kuhn, Ivica Letunic, Lars Juhl Jensen, and Peer Bork. The sider database of drugs and side effects. *Nucleic acids research*, 44(D1):D1075–D1079, 2016.

[22] Kaitlyn M Gayvert, Neel S Madhukar, and Olivier Elemento. A data-driven approach to predicting successes and failures of clinical trials. *Cell chemical biology*, 23(10):1294–1301, 2016.

[23] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[24] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[25] Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388, 2019.

[26] Chengqiang Lu, Qi Liu, Chao Wang, Zhenya Huang, Peize Lin, and Lixin He. Molecular property prediction: A multilevel quantum interactions modeling perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1052–1060, 2019.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[28] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform bad for graph representation? *arXiv preprint arXiv:2106.05234*, 2021.

[29] Guy W Bemis and Mark A Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of medicinal chemistry*, 39(15):2887–2893, 1996.

[30] Greg Landrum et al. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling, 2013.

[31] Stefan Chmiela, Huziel E Sauceda, Igor Poltavsky, Klaus-Robert Müller, and Alexandre Tkatchenko. sgdml: Constructing accurate and data efficient molecular force fields using machine learning. *Computer Physics Communications*, 240:38–45, 2019.

[32] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. Se (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *arXiv preprint arXiv:2101.03164*, 2021.

[33] Stefan Chmiela, Huziel E Sauceda, Klaus-Robert Müller, and Alexandre Tkatchenko. Towards exact molecular dynamics simulations with machine-learned force fields. *Nature communications*, 9(1):1–10, 2018.

[34] Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. *arXiv preprint arXiv:2102.09844*, 2021.

[35] Shengchao Liu, Mehmet F Demirel, and Yingyu Liang. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *Advances in neural information processing systems*, 32, 2019.

[36] Bharath Ramsundar, Steven Kearnes, Patrick Riley, Dale Webster, David Konerding, and Vijay Pande. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*, 2015.

[37] Steven Kearnes, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick Riley. Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design*, 30(8):595–608, 2016.

[38] Kristof T Schütt, Pieter-Jan Kindermans, Huziel E Sauceda, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *arXiv preprint arXiv:1706.08566*, 2017.

[39] Zhaoping Xiong, Dingyan Wang, Xiaohong Liu, Feisheng Zhong, Xiaozhe Wan, Xutong Li, Zhaojun Li, Xiaomin Luo, Kaixian Chen, Hualiang Jiang, et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *Journal of medicinal chemistry*, 63(16):8749–8760, 2019.

[40] Alan R Katritzky, Minati Kuanar, Svetoslav Slavov, C Dennis Hall, Mati Karelson, Iiris Kahn, and Dimitar A Dobchev. Quantitative correlation of physical and chemical properties with chemical structure: utility for prediction. *Chemical reviews*, 110(10):5714–5789, 2010.

[41] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

## Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The data is public dataset. The instructions are listed in Section **??**. The code will be open-sourced on GitHub
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]