LEARNING OVER MOLECULAR CONFORMER ENSEMBLES: DATASETS AND BENCHMARKS

Yanqiao Zhu UCLA **Jeehyun Hwang** UCLA **Keir Adams** MIT **Zhen Liu** CMU **Bozhao Nan** Notre Dame
Brock Anton Stenfors Notre Dame **Yuanqi Du** Cornell **Jatin Chauhan** UCLA **Olaf Wiest** Notre Dame
Olexandr Isayev CMU **Connor W. Coley** MIT **Yizhou Sun** UCLA **Wei Wang** UCLA

Primary contact: yzhu@cs.ucla.edu
Project homepage: https://github.com/SXKDZ/MARCEL

ABSTRACT

Molecular Representation Learning (MRL) has proven impactful in numerous biochemical applications such as drug discovery and enzyme design. While Graph Neural Networks (GNNs) are effective at learning molecular representations from a 2D molecular graph or a single 3D structure, existing works often overlook the flexible nature of molecules, which continuously interconvert across conformations via chemical bond rotations and minor vibrational perturbations. To better account for molecular flexibility, some recent works formulate MRL as an ensemble learning problem, focusing on explicitly learning from a set of conformer structures. However, most of these studies have limited datasets, tasks, and models. In this work, we introduce the first MoleculAR Conformer Ensemble Learning (MARCEL) benchmark to thoroughly evaluate the potential of learning on conformer ensembles and suggest promising research directions. MARCEL includes four datasets covering diverse molecule- and reaction-level properties of chemically diverse molecules including organocatalysts and transition-metal catalysts, extending beyond the scope of common GNN benchmarks that are confined to drug-like molecules. In addition, we conduct a comprehensive empirical study, which benchmarks representative 1D, 2D, and 3D MRL models, along with two strategies that explicitly incorporate conformer ensembles into 3D models. Our findings reveal that direct learning from an accessible conformer space can improve performance on a variety of tasks and models.

1 INTRODUCTION

Recent years have seen the emergence of Molecular Representation Learning (MRL) as a promising approach for modeling molecules with machine learning. In the typical formulation, MRL maps discrete molecular objects to continuous features in a data-driven manner, encoding complex chemical structures into representation vectors that can subsequently be utilized in different downstream tasks. In particular, MRL now underpins a variety of biochemical applications spanning molecular property prediction to the design of novel drug candidates [1–3].

Traditional approaches often encode chemical compounds with fingerprints, such as extended-connectivity fingerprints [4, 5], which indicate the existence of certain substructures as binary bits in a fixed-length sequence. Such line-based representations are concise and efficient, but have limited expressive power and have difficulty in capturing 3D structural information such as bonding geometries and global shapes, which can be important for analyzing molecular properties and chemical reactivity [6, 7]. Recently, Graph Neural Networks (GNNs) have become an increasingly popular method of learning molecular representations by treating molecules as graph-structured objects. Existing GNN models for MRL can be broadly classified into two categories: 2D topological models [8–11] and 3D geometric models [12–17]. 2D GNNs typically model the molecular connectivity as a flat 2D graph with atoms as nodes and bonds as edges, learning representations of chemical environments by iteratively passing messages between neighboring atoms. Although powerful in the absence of structural information, 2D GNNs may fail to capture key conformational effects or stereochemical properties like chirality [18, 19], which is critical for modeling molecular interactions in areas such as

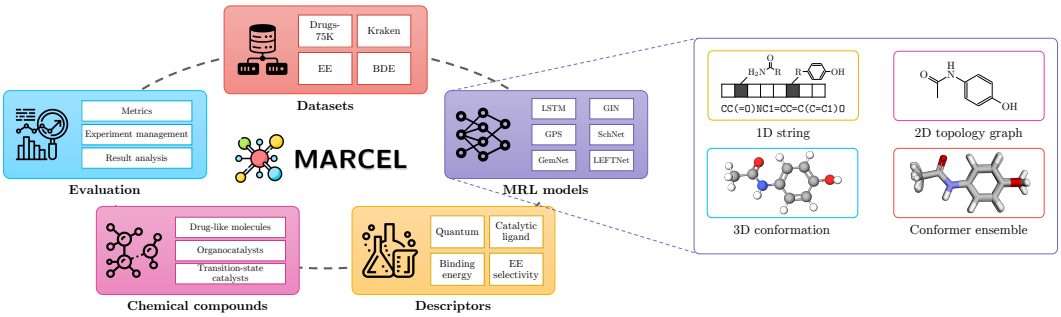

Figure 1: We present a MARCEL benchmark that comprehensively evaluates the potential of learning on conformer ensembles across a diverse set of molecules, datasets, and models.

drug design or chemical catalysis. Conversely, 3D GNNs are designed to model molecular conformers (conformations), which describe the structure of molecules in 3D space. Thus, these models have found widespread adoption for modeling electronic properties, predicting conformer energies and forces, and scoring interactions between ligands and proteins, amongst other applications.

In almost all applications, benchmarks, and demonstrations, 3D GNN models focus on encoding *individual* conformer structures. It is critical to recognize that in reality molecules are not rigid, static objects; rather, thermodynamically-permissible rotations of chemical bonds, small vibrational motions, and dynamic intermolecular interactions cause molecules to continuously convert between different conformations [20]. As a consequence, many experimentally observable chemical properties depend on the full distribution of thermodynamically-accessible conformers. For example, a molecule needs to be arranged into a particular pose to bind to a target protein, and this binding conformation changes depending on the dynamic interaction between the molecule and the target [21]. Also, it is often challenging to determine *a priori* the conformers that predominantly contribute to molecular properties without doing prohibitively expensive simulations. Therefore, a natural question arises: can we leverage the *collective* power of many different conformer structures lying on the local minima of the potential energy surface, also known as the *conformer ensemble*, to improve MRL models?

As shown by the empirical evidence from various studies, learning from an explicit conformer ensemble can prove to be advantageous for many tasks, including property and energy prediction [22–24], key conformer pose identification [25], and RNA sequence design [26]. However, these studies have been mostly confined to small-scale datasets, a limited set of tasks, and a restricted set of model architectures. As a result, it remains unclear (1) to what extent 2D GNNs can implicitly model molecular flexibility and (2) whether the *explicit* encoding of conformer ensembles can improve the performance of 3D models that traditionally encode only one single conformer.

In this paper, we present the first MolecuLAR Conformer Ensemble Learning (MARCEL) benchmark. It covers a diverse range of chemical space (Figure 1), which focuses on four chemically-relevant tasks for both molecules and reactions, with an emphasis on Boltzmann-averaged properties of conformer ensembles computed at the Density-Functional Theory (DFT) level. Our datasets encompass a variety of compounds with high-quality conformers, including organocatalysts and transition-metal catalysts, extending beyond the scope of conventional GNN benchmarks which are often restricted to drug-like molecules. Moreover, we implement a benchmark suite that enables extensive empirical studies across representative 1D, 2D, and 3D models. We further explore the advantages of leveraging conformer ensembles through two straightforward strategies: (1) augmenting training samples by randomly selecting one conformer from the ensemble for each molecule and (2) applying an explicit multi-instance ensemble learning layer, which aggregates individual conformer embeddings.

Our experimental results confirm the potential effectiveness of incorporating conformer ensembles in MRL, highlighting the improvements over conventional single-conformation 3D networks. However, it is important to understand the heterogeneity of outcomes based on different dataset characteristics, task objectives, and model choices. Our investigation yields three key findings: (1) Leveraging molecular conformers by incorporating explicit set encoders, as a part of conformer ensemble learning strategies, can improve single-conformer 3D MRL models performance. (2) Data augmentation through conformer sampling may offer potential benefits, evidenced by improved results in the BDE dataset, suggesting a method to increase model robustness against imprecise structures. (3) Model selection for MRL depends on dataset sizes and tasks, with traditional 1D fingerprints and 2D models preferred for smaller datasets and 3D models for larger or reaction-focused tasks.

2 PROBLEM FORMULATION

We represent a 2D molecular graph as a tuple $\mathsf{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{X}, \boldsymbol{W})$, where $\mathcal{V} = \{v_i\}_{i=1}^{|\mathcal{V}|}$ is the node set with each node corresponding to an atom, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set representing chemical bonds as edges between nodes. Further, $\boldsymbol{X} \in \mathbb{R}^{d_v \times |\mathcal{V}|}$ contains vector attributes for each node, and $\boldsymbol{W} \in \mathbb{R}^{d_w \times |\mathcal{E}|}$ contains attributes for each edge. When modeling chemical reactions, we represent a molecule-molecule complex as a pair of graphs $(\mathsf{G}_1, \mathsf{G}_2)$. In this case, the conformation describes the combined structure of the interacting molecules. For a given molecule or molecular complex, we assume that its geometry can be effectively characterized by a representative set of discrete, sampled conformers from the thermodynamically-accessible conformer distribution. Formally, this set can be denoted as $\mathcal{C} = \{\boldsymbol{C}_i\}_{i=1}^{|\mathcal{C}|}$, where $\boldsymbol{C}_i \in \mathbb{R}^{|\mathcal{V}| \times 3}$ represents one conformer structure in 3D space. In reality, the conformer distribution is continuous; $\mathcal{C}$ in our study contains representative samples of the infinite set. Each conformer in the sampled ensemble is associated with a statistical weight given by

$$p_i = \frac{\exp\left(-\frac{e_i}{k_B T}\right)}{\sum_j \exp\left(-\frac{e_j}{k_B T}\right)},$$

which corresponds to its probability under experimental conditions. Here, e_i is the energy of the conformer $\boldsymbol{C}_i$, k_B is the Boltzmann constant, and T is the temperature. Notably, p_i is not prior information to the models analyzed in this benchmark. Rather, we use a discrete approximation of p_i to compute the ground-truth labels for our regression tasks.

3 DATASETS AND TASKS

MARCEL contains four small-to-large-scale datasets involving nine regression tasks with considerably diverse chemistry. Drugs-75K and Kraken focus on molecular properties, while EE and BDE focus on reaction-centric properties. MARCEL includes molecules with high structural flexibility, evidenced by an average number of rotatable bonds exceeding 5. Table 1 summarizes the datasets.

Drugs-75K is a subset of the GEOM-Drugs [27] dataset, which includes 75,099 molecules with at least 5 rotatable bonds. For each molecule, Auto3D [28] is used to generate and optimize the conformer ensembles and AIMNet-NSE [29] is used to calculate three important quantum chemical descriptors: ionization potential, electron affinity, and electronegativity [30]. Note that Auto3D and AIMNet-NSE achieve DFT-level accuracy but are much more efficient [21, 31, 32].

- Ionization Potential (IP) is the minimum energy required to remove an electron from a neutral atom or molecule to form a positively charged ion (cation): $\text{IP} = E_{\text{cation}} - E_{\text{neutral}}$.
- Electron Affinity (EA) denotes the energy change associated with the addition of an electron to a neutral atom or molecule to form a negatively charged ion (anion): $\text{EA} = E_{\text{neutral}} - E_{\text{anion}}$.
- Electronegativity (χ) measures the tendency of an atom to attract a bonding pair of electrons:

$$\chi = -\left(\frac{\partial E}{\partial N}\right).$$

E_{cation}, E_{neutral}, and E_{anion} are the electronic energy of the positively charged, neutral, and negatively charged molecules, respectively. E and N are the energy and the number of electrons, respectively.

The tasks are to predict the Boltzmann-averaged value of each property across the conformer ensemble $\langle y \rangle_{k_B} = \sum_{\boldsymbol{C}_i \in \mathcal{C}} p_i y_i$, where y_i is a conformer-specific property. We are given each $\boldsymbol{C}_i$, and the goal is to predict $\langle y \rangle_{k_B}$ from the molecular graph G, a single conformer $\boldsymbol{C}_i \in \mathcal{C}$, or the set $\mathcal{C}$.

Kraken [33] is a dataset of 1,552 monodentate organophosphorus (III) ligands along with their DFT-computed conformer ensembles. In this study, we consider four 3D ligand descriptors exhibiting significant variance among conformers: Sterimol B_5, Sterimol L, buried Sterimol B_5, and buried Sterimol L. These descriptors quantify the steric features of each ligand in units of Å and are often employed for Quantitative Structure-Activity Relationship (QSAR) modeling in catalyst design.

As in the Drugs-75K tasks, the goal is to predict the Boltzmann-averaged value of each property across the conformer ensemble from the molecular graph G, a single conformer $\boldsymbol{C}_i \in \mathcal{C}$, or the set $\mathcal{C}$.

Table 1: Statistics of the four datasets. The numbers of heavy atoms and rotatable bonds ("rot. bonds") are averaged per conformer.

Dataset	# Molecules	# Conformers	# Heavy atoms	# Rot. bonds	# Targets	Atomic species
Drugs-75K	75,099	558,002	30.56	7.53	3	H, C, N, O, F, Si, P, S, Cl
Kraken	1,552	21,287	23.70	9.05	4	H, B, C, N, O, F, Si, P, S, Cl, Fe, Se, Br, Sn, I
Dataset	# Reactions	# Conformers	# Heavy atoms	# Rot. bonds	# Targets	Atomic species
EE	872	Pro-R: 14,807 Pro-S: 13,999	59.32	18.57	1	H, C, N, O, F, P, Cl, Br, Rh
BDE	5,915	Ligand: 73,834 Complex: 40,264	29.62 32.38	6.99 6.99	1	H, C, N, O, F, P, Cl, Ni, Cu, Br, Pd, Ag, Pt, Au

EE [34] is a dataset of 872 catalyst-substrate pairs involving 253 Rhodium (Rh)-bound atropisomeric catalysts derived from chiral bisphosphine, with 10 enamides as substrates. The dataset includes conformations of catalyst-substrate transition state complexes in two separate pro-S and pro-R configurations. The task is to predict the Enantiomeric Excess (EE) of the chemical reaction involving the substrate, defined as the absolute ratio between the concentration of each enantiomer in the product distribution. This dataset is generated with Q2MM, which automatically generates Transition State Force Fields (TSFFs) in order to simulate the conformer ensembles of each prochiral transition state complex. EE can then be computed from the conformer ensembles by Boltzmann-averaging the activation energies for the competing transition states [34, 35].

Unlike properties in Drugs-75K and Kraken, EE depends on the conformer ensembles of *each* pro-R and pro-S complex. The goal is to predict EE from the graphs of the catalyst and substrate $(\mathsf{G}_{\text{cat}}, \mathsf{G}_{\text{sub}})$, a conformer $\boldsymbol{C}_i^{(\text{R})} \in \mathcal{C}^{(\text{R})}$ and $\boldsymbol{C}_i^{(\text{S})} \in \mathcal{C}^{(\text{S})}$ for each complex, or the ensembles $\mathcal{C}^{(\text{R})}$ and $\mathcal{C}^{(\text{S})}$.

BDE [36] is a dataset containing 5,915 organometallic catalysts ML_1L_2 consisting of a metal center (M = Pd, Pt, Au, Ag, Cu, Ni) coordinated to two flexible organic ligands (L_1 and L_2), each selected from a 91-membered ligand library. The data includes conformations of each unbound catalyst, as well as conformations of the catalyst when bound to ethylene and bromide after oxidative addition with vinyl bromide. Each catalyst has an electronic binding energy, computed as the difference in the minimum energies of the bound-catalyst complex and unbound catalyst, following the DFT-optimization of their respective conformer ensembles. Although the binding energies are computed via DFT, the conformers provided for modeling are initially generated with Open Babel [37] followed by further geometry optimization, which ensures that the 3D structures are likely to be the global minimum energy conformers at the force field level [36]. Note that obtaining DFT-optimized conformers for BDE is not feasible given the significant computational cost. Therefore, this realistically represents the setting in which precise conformer ensembles are unknown at inference.

The task is to predict the binding energy from the graphs of the unbound and bound catalyst, sampled conformers $\boldsymbol{C}_i^{(\text{unbound})} \in \mathcal{C}^{(\text{unbound})}$ and $\boldsymbol{C}_i^{(\text{bound})} \in \mathcal{C}^{(\text{bound})}$, or the ensembles $\mathcal{C}^{(\text{unbound})}$ and $\mathcal{C}^{(\text{bound})}$.

Dataset Preparation. We implement several preprocessing steps to ensure the quality and validity of our datasets and facilitate their integration into machine learning models.

- **Conformer deduplication.** To eliminate redundant conformers in each ensemble $\mathcal{C}$, we first align every pair of conformers using RDKit [38], accounting for symmetric atom permutations. Subsequently, we employ Butina clustering [39] based on the Root Mean Square Deviation (RMSD) values derived from conformer alignment. Within each cluster, we select the conformer with the lowest energy. Note that Boltzmann-averaged regression labels are computed *before* deduplication.
- **Selection of molecules.** We focus on modeling flexible molecules, for which conformer ensemble learning may be relevant to capture their properties. Hence, we only retain molecules with more than 5 rotatable bonds. We also remove molecules with missing 3D geometries or 2D graphs.

4 Benchmarking Molecular Representation Learning Models

The representation of molecular data is crucial for applying machine learning models to problems in chemistry and biology. These representations typically include 1D strings, 2D topological graphs, and 3D geometric graphs. For a comprehensive benchmark for MRL models, our MARCEL includes a diverse representative selection of models for each of the aforementioned molecular representations.

In this section, we provide an overview of these models and describe how they are tailored to our tasks. We also introduce two strategies of explicitly encoding conformer ensembles using 3D models.

4.1 1D MODELS

Our 1D baselines include Random Forest [40] models operating on molecular fingerprints [38, 41, 42]. Fingerprints convert a molecular graph into a bit array indicating the presence of chemical substructures and are widely used for cheminformatics and QSAR modeling in the low-data regime. Additionally, we include Long Short-Term Memory (LSTM) [43] and Transformer [44] models, popular sequence-based neural network architectures, operating on SMILES strings. For the BDE and EE datasets, we concatenate the SMILES of each molecule or complex with a "." symbol and use a single sequence encoder. Further details on model implementations can be found in Appendix B.1.

4.2 2D GRAPH NEURAL NETWORKS

We employ four widely-used GNN models as 2D baseline methods, including Graph Isomorphism Network (GIN) [45], GIN with Virtual Node (GIN-VN) [46], ChemProp [47], and GraphGPS [48]. GIN is a commonly-used model with strong representation ability. GIN-VN augments the vanilla GIN by incorporating a virtual node to aggregate the features of all nodes in the graph, thereby capturing global information more effectively. ChemProp is a directed message passing GNN designed specifically for molecular property prediction. GraphGPS is a Transformer-like [44] model specifically tailored for graph-structured data, which is able to capture long-range relationships.

Following OGB protocols [46], we employ a diverse set of atomic features such as aromaticity and hybridization for nodes, as well as bond features like ring information for edges (Appendix B.2). For the EE and BDE datasets, we employ a two-tower architecture with two separate 2D GNN models: for EE, since both pro-S and pro-R complexes share the same 2D graph, we leverage two separate GNNs to encode the catalyst and substrate; for BDE, we also encode the unbound and bound catalysts separately. We then concatenate these together to obtain the system-level embeddings.

4.3 3D GRAPH NEURAL NETWORKS

We include six representative 3D GNNs that encompass diverse modeling perspectives. SchNet [12], an E(3)-invariant network, models spatial interactions by encoding pairwise interatomic distance. DimeNet++ [13], another E(3)-invariant model, uses directional message passing that embeds angles between triplets of atoms in order to enhance geometric expressivity. GemNet [14], an SE(3)-invariant model, utilizes a unique attention mechanism and dihedral angles between four atoms to model atomic interactions. PaiNN [15], initially developed to predict tensorial properties and molecular spectra, incorporates rotational equivariance into its message passing framework. ClofNet [16], an SE(3)-equivariant model that improves the popular EGNN [49], uses complete local frames for each atom, effectively capturing 3D atomistic structures while preserving invariance and equivariance. LEFTNet [17], based on ClofNet, introduces Local Substructure Encoding (LSE) and Frame Transition Encoding (FTE) to enhance the model expressivity via scalarization and tensorization.

We use atom types as the sole atom features for the 3D models. For both training and inference on Drug-75K, Kraken, and EE datasets, all the single-conformer 3D models encode the lowest-energy conformer of each conformer ensemble, which has the largest Boltzmann weight and hence provides the strongest model. Since imprecise conformers are encoded for the BDE task, we use a fixed, randomly sampled conformer for each unbound- and bound-catalyst during training and inference.

The 3D models also employ a two-tower architecture for the EE and BDE datasets. Two separate 3D GNNs are used to encode representations for each pro-S and pro-R complex in EE, or for each catalyst and bound complex in BDE, which are then concatenated to form the final representations.

We note that although using the lowest-energy conformer will yield the strongest performance, this setting can be unrealistic: it is often not possible to identify the lowest energy conformer without searching the entire conformer space. The lowest energy conformer can also depend on the force field used for geometry optimization, which may neglect experimental conditions such as solvents.

4.4 INCORPORATING CONFORMER ENSEMBLES INTO MOLECULAR REPRESENTATIONS

3D geometric models primarily focus on learning representations from individual 3D structures. Although some models preserve global symmetries such as SE(3)-equivariance, these models do not learn representations that capture conformational flexibility which is caused by internal degrees of freedom such as bond rotations. Here, we describe two straightforward strategies that model conformational flexibility by explicitly leveraging conformer ensembles.

4.4.1 STRATEGY 1: TRAINING-TIME DATA AUGMENTATION VIA CONFORMER SAMPLING

A direct approach to modeling conformer flexibility is to simply enrich the training data by randomly sampling a conformer from the ensemble during each training epoch. Formally, for a given molecule G and its conformer ensemble $\mathcal{C}$, we randomly select a conformer with uniform probability $p = 1/|\mathcal{C}|$ while using the same training label for each conformer. Note that during inference, the lowest-energy conformer is used to evaluate the model performance. This strategy aligns with learning representations invariant to conformational changes, thus implicitly encoding the flexibility of molecular structures, and has been shown to be useful for learning chirality-sensitive 3D representations [19]. When conformer ensembles are available, the strategy is computationally efficient as it maintains the same complexity as the base 3D model. Unlike the other ensemble methods, this strategy can be used if conformer ensembles are only available at training time. In Appendix C, we evaluate two alternative scenarios where conformer ensembles are also available during evaluation.

4.4.2 STRATEGY 2: ENSEMBLE LEARNING WITH EXPLICIT SET ENCODERS

The second strategy utilizes a set encoder to simultaneously encode the entire conformer ensemble $\mathcal{C}$ at both training and inference time. Inspired by the multi-instance learning framework [50–52], this strategy first employs 3D GNNs to generate individual conformer embeddings and then aggregates these embeddings using a set encoder, as illustrated in Figure 2.

Formally, for each conformer $\boldsymbol{C}_i \in \mathcal{C}$, we obtain its corresponding embedding $\boldsymbol{z}_i = f(\mathsf{G}, \boldsymbol{C}_i) \in \mathbb{R}^d$, where f is a single-conformer 3D model and d is the embedding dimension. Note that the embedding $\boldsymbol{z}$ is a (3D) graph-level representation resulting from a pooling function over the node-level embeddings after message passing. To further aggregate these embeddings $\mathcal{Z} = \{\boldsymbol{z}_i\}_{i=1}^{|\mathcal{C}|}$ into a single molecular representation, we consider the following three set encoders:

- **Mean pooling** simply computes the mean of all the conformer embeddings:

$$\boldsymbol{s}^{\text{MEAN}} = \frac{1}{|\mathcal{C}|} \sum_{i=1}^{|\mathcal{C}|} \boldsymbol{z}_i. \tag{1}$$

- **DeepSets** [53] utilizes a permutation-invariant function to process a set of inputs. It first applies a MultiLayer Perceptron (MLP) h to each conformer embedding and then aggregates the transformed embeddings using sum pooling followed by another MLP g:

$$\boldsymbol{s}^{\text{DS}} = g\left(\sum_{i=1}^{|\mathcal{C}|} h(\boldsymbol{z}_i)\right). \tag{2}$$

  This method retains more discernible information from individual embeddings compared to mean pooling at a cost of two non-linear functions.

- **Self-attention** [54] further computes a weighted sum of the embeddings, where the weights are obtained by applying a softmax function to the dot product of the embeddings:

$$\boldsymbol{s}^{\text{ATT}} = \sum_{i=1}^{|\mathcal{C}|} \boldsymbol{c}_i, \quad \text{where } \boldsymbol{c}_i = g\left(\sum_{j=1}^{|\mathcal{C}|} \alpha_{ij} h(\boldsymbol{z}_j)\right), \ \alpha_{ij} = \frac{\exp((\boldsymbol{W}h(\boldsymbol{z}_i))^\top (\boldsymbol{W}h(\boldsymbol{z}_j)))}{\sum_{k=1}^{|\mathcal{C}|} \exp((\boldsymbol{W}h(\boldsymbol{z}_i))^\top (\boldsymbol{W}h(\boldsymbol{z}_k)))}. \tag{3}$$

  Here, $\boldsymbol{W} \in \mathbb{R}^{d \times d}$ is a learnable weight matrix. This approach can capture conformer interactions.

By employing these set encoders, we can learn a model that is more sensitive to the full range of conformer variations present in the ensemble. After obtaining the ensemble embeddings, we further apply a linear projection head to generate the final prediction.

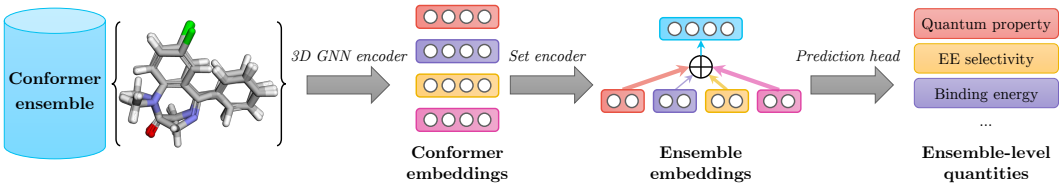

Figure 2: Conformer ensemble learning with explicit set encoders (Strategy 2). Individual conformer embeddings are first obtained via 3D GNN encoders. Then, a set encoder is employed to aggregate conformer embeddings. Finally, a linear projection head is used to generate the prediction.

5 Experiments

5.1 Experimental Configurations

Each dataset is partitioned randomly into three subsets: 70% for training, 10% for validation, and 20% for test. Each model is trained over 2,000 epochs using the Adam optimizer [55] with early stopping triggered if there is no improvement on the training loss over 200 epochs. For all nine regression targets, experiments are repeated three times, and the results reported correspond to the model that performs best on the validation set in terms of Mean Absolute Error (MAE).

The Boltzmann-averaged targets are computed over all available conformers. For ensemble learning models, we cap the number of encoded conformers per molecule to a maximum of 20, which empirically improves training stability and leads to better performance. To ensure a fair comparison, the hidden dimension size is uniformly set to 128 for all models. Other settings mostly follow the original configurations as described in the respective papers. We specify all hyperparameters and describe experimental environments in Appendix B.3.

5.2 Results and Analysis

We summarize the performance of the 1D, 2D, and 3D MRL models and the best results from ensemble learning strategies on 3D models in Table 2. Figure 3 reports the *performance changes* in Mean Absolute Error (MAE) for each 3D model when applying the ensemble learning strategies. The raw performance data with standard deviation and the parameter size of each model can be found in Appendix D. In summary, although performance varies across the datasets, tasks, and models, the ensemble learning strategies improve upon 3D models that only encode one conformer in 48 out of 54 experiments across 9 tasks and 6 base models, demonstrating the effectiveness of conformer ensemble learning. Our analysis leads to the following key observations.

Observation 1. The conformer ensemble learning strategy with explicit set encoders frequently yields improved performance.

Figure 3 indicates that encoding conformer ensembles can substantially reduce test error, achieving improvements in 108 experiments across all 9 tasks, 6 base models, and 3 set encoders, most notably on the tasks in the smaller-sized Kraken dataset. This, however, does not always extend to larger datasets like Drugs-75K. We conjecture that for Drugs-75K, the computational burden of encoding all conformers in each ensemble alters the learning dynamics of the underlying model, making training more challenging. A similar finding was reported by Axelrod and Gómez-Bombarelli [23].

Among the three set encoders, DeepSets consistently demonstrates significant improvements in 42 out of 54 experiments across 9 tasks and 6 base 3D models. We conjecture that this superior performance is due to its ability of effectively modeling set objects at a relatively minor computational overhead of two non-linear transformations. On the other hand, the simple mean pooling approach loses discriminative power across the conformers in the ensemble, resulting in inferior performance. It is also noteworthy that the attention models exhibit mixed results compared to the vanilla 3D models, despite theoretically being the most powerful set encoders. This inconsistency might be attributable to the computational intricacy of the self-attention layer, which models the pairwise relationship among conformers in each ensemble and hence could require more sophisticated training strategies. Future research should consider developing better neural architectures that are specifically designed to more efficiently encode structural information from conformer ensembles.

Table 2: Performance of 1D, 2D, and 3D baseline MRL models and the best results from ensemble learning strategies on 3D GNNs. The metric used is the Mean Absolute Error (MAE, ↓). The **bold** indicates the best-performing model, while underlined denotes the second-best.

Category	Model	Drugs-75K			Kraken				EE	BDE
		IP	EA	χ	B_5	L	$BurB_5$	BurL		
1D	Random forest	0.4987	0.4747	0.2732	0.4760	0.4303	0.2758	0.1521	61.2963	3.0335
	LSTM	0.4788	0.4648	0.2505	0.4879	0.5142	0.2813	0.1924	64.0088	2.8279
	Transformer	0.6617	0.5850	0.4073	0.9611	0.8389	0.4929	0.2781	62.0816	10.0771
2D	GIN	0.4354	0.4169	0.2260	0.3128	0.4003	0.1719	0.1200	62.3065	2.6368
	GIN+VN	0.4361	0.4169	0.2267	0.3567	0.4344	0.2422	0.1741	62.3815	2.7417
	ChemProp	0.4595	0.4417	0.2441	0.4850	0.5452	0.3002	0.1948	61.0336	2.6616
	GraphGPS	0.4351	0.4085	0.2212	0.3450	0.4363	0.2066	0.1500	61.6251	2.4827
3D	SchNet	0.4394	0.4207	0.2243	0.3293	0.5458	0.2295	0.1861	17.7421	2.5488
	DimeNet++	0.4441	0.4233	0.2436	0.3510	0.4174	0.2097	0.1526	14.6414	**1.4503**
	GemNet	0.4069	0.3922	**0.1970**	0.2789	0.3754	0.1782	0.1635	18.0338	1.6530
	PaiNN	0.4505	0.4495	0.2324	0.3443	0.4471	0.2395	0.1673	20.2359	2.1261
	ClofNet	0.4393	0.4251	0.2378	0.4873	0.6417	0.2884	0.2529	33.9473	2.6057
	LEFTNet	0.4174	0.3964	0.2083	0.3072	0.4493	0.2176	0.1486	19.7974	1.5328
Best Ensemble Strategy	SchNet	0.4452	0.4232	0.2243	0.2704	0.4322	0.2024	0.1443	14.2238	1.9737
	DimeNet++	0.4126	0.3944	0.2267	0.2630	0.3468	0.1783	0.1185	12.0259	1.4741
	GemNet	**0.4066**	**0.3910**	0.2027	0.2313	**0.3386**	**0.1589**	**0.0947**	**11.6142**	1.6059
	PaiNN	0.4466	0.4269	0.2294	**0.2225**	0.3619	0.1693	0.1324	13.5570	1.8744
	ClofNet	0.4280	0.4033	0.2199	0.3228	0.4485	0.2178	0.1548	13.9647	2.0106
	LEFTNet	0.4149	0.3953	0.2069	0.2644	0.3643	0.2017	0.1386	18.4189	1.5276

Observation 2. Sampling conformers at training time can improve performance, especially on imprecise conformer structures.

We observe that data augmentation improves performance on 34 experiments, especially on the challenging BDE dataset, where the other ensemble learning strategies often do not help. Note that the conformers in the BDE dataset originate from Open Babel, as opposed to the golden-standard DFT-level conformers present in other datasets. This suggests that training with randomly sampled conformers might offer robustness to noise in the imprecise structures. On other tasks, randomly sampling the conformers at each epoch may help the model learn an invariance to conformational changes, but does not always increase performance for all 3D models. This might be because the sampling probability is uniform across the entire conformer set, which does not respect the underlying Boltzmann weight of each conformer. In future work, it may be worthwhile to investigate whether more physics-informed sampling strategies could lead to more consistent performance gains.

Observation 3. No model consistently outperforms the rest, with substantial task dependencies.

The results in Table 2 suggest that no single model outperforms the others across all tasks. Of the 1D models, LSTM outperforms Random Forest and Transformer models on Drugs-75K and BDE, demonstrating the effectiveness of SMILES-based representations of molecules on large-scale datasets. For small datasets such as Kraken and EE, Random Forests outperform sequence models at a lower computational cost, indicating that traditional models are superior in the low-data regime.

Amongst 2D models, GIN delivers the best performance on four tasks compared to all other models; GraphGPS also demonstrates strong performance on several tasks (B_5, L, and BurL). Surprisingly, the 2D models are also competitive with some 3D models on the large-scale Drugs-75K tasks. This is possibly due to the fact that the electronic properties in Drugs-75K are not as sensitive to conformational changes, thus explicitly modeling the structures in 3D may not be necessary. However, all 2D models perform worse as compared to the 3D models in the reaction datasets EE and BDE, indicating the important role of spatial interactions in determining reaction-related properties.

For 3D models, GemNet and LEFTNet excel in IP, EA, and χ. The complexity of these two equivariant models may especially benefit from the large dataset size of Drugs-75K. For Kraken and the two reaction datasets, DimeNet++ — an invariant model — achieves promising performance, suggesting that highly-complex 3D models may be less useful for chemical applications with small-to-medium sized datasets. On EE, we observe that 3D models remarkably outperform 1D and 2D models, likely because enantioselectivity depends on subtle spatial interactions. When predicting binding energies, using 3D models also leads to modest improvements.

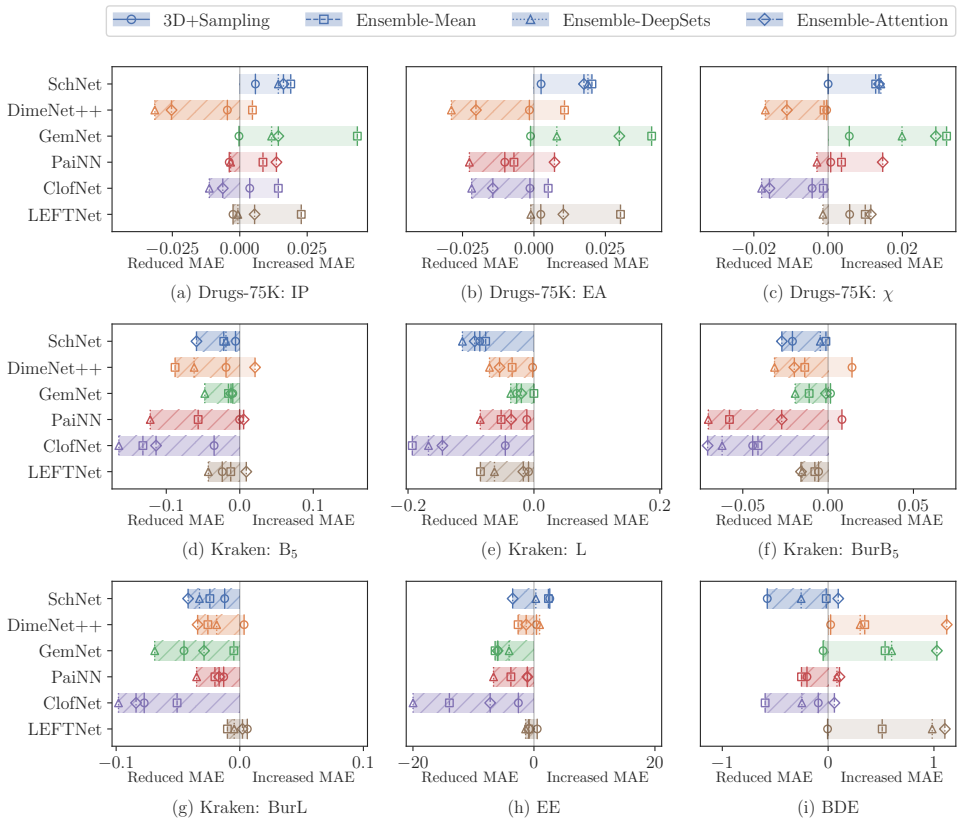

Figure 3: *Performance changes* of four conformer ensemble learning strategies on the basis of six 3D graph models. Here, negative values (marked in hatch patterns) denote *reduced* Mean Absolute Error (MAE), signifying a performance improvement due to the incorporation of conformer ensembles.

Overall, model performance varies substantially across tasks, even within the same dataset, emphasizing the diversity of the tasks in MARCEL. Generally, 1D and 2D models perform well on small-scale molecular datasets, while 3D models excel on large datasets and reaction-centric tasks. MARCEL also highlights the benefits of explicitly encoding multiple conformers to improve MRL.

6 Discussions and Conclusions

In this work, we present the first MoleculAR Conformer Ensemble Learning benchmark (MARCEL) to evaluate the potential of learning from a set of conformer structures. Through two conformer ensemble learning strategies, we discover performance improvements across various tasks. However, there are some limitations that require further consideration. First, our studied ensemble learning strategies do not universally improve performance across all tasks and datasets. This highlights the need for more tailored approaches that integrate with domain expertise to better model specific tasks and datasets of practical interest. Second, the computational cost of encoding all conformers within the ensembles, especially for larger datasets, suggests the need to further study the trade-offs between model complexity and efficiency. Finally, our datasets only contain regression tasks and do not cover all of the relevant chemical space, which might limit the generalization of our experimental findings.

Despite these challenges, we envision that our work will prompt further research in the geometric deep learning community on how to make use of conformer ensembles for molecular property prediction. For instance, future research could explore new model architectures that can efficiently encode ensemble-level information or more sophisticated conformer sampling strategies. We also hope that our work will stimulate collaborative research across the machine learning and chemistry fields, with the ultimate goal of pushing the boundaries of predictive molecular modeling and aligning algorithmic advancements with the practical needs of the chemistry community.

ACKNOWLEDGEMENTS

This work is supported by NSF Center for Computer Assisted Synthesis (2202693).

REFERENCES

[1] Jun Xia, Yanqiao Zhu, Yuanqi Du, and Stan Z. Li. A Systematic Survey of Chemical Pre-trained Models. 2023. 1

[2] Oliver Wieder, Stefan Kohlbacher, Mélaine Kuenemann, Arthur Garon, Pierre Ducrot, Thomas Seidel, and Thierry Langer. A Compact Review of Molecular Property Prediction with Graph Neural Networks. *Drug Discov. Today Technol.*, 37:1–12, 2020. 1

[3] W. Patrick Walters and Regina Barzilay. Applications of Deep Learning in Molecule Generation and Molecular Property Prediction. *Acc. Chem. Res.*, 54(2):263–270, 2021. 1

[4] H. L. Morgan. The Generation of a Unique Machine Description for Chemical Structures — A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.*, 5(2):107–113, 1965. 1

[5] Robert C. Glem, Andreas Bender, Catrin H. Arnby, Lars Carlsson, Scott Boyer, and James Smith. Circular Fingerprints: Flexible Molecular Descriptors with Applications from Physical Chemistry to ADME. *IDrugs*, 9(3):199–204, 2006. 1

[6] G. Skoraczyński, P. Dittwald, B. Miasojedow, S. Szymkuć, E. P. Gajewska, B. A. Grzybowski, and A. Gambin. Predicting the Outcomes of Organic Reactions via Machine Learning: Are Current Descriptors Sufficient? *Sci. Rep.*, 7(1):1–9, 2017. 1

[7] Zhen Liu, Yurii S. Moroz, and Olexandr Isayev. The Challenge of Balancing Model Sensitivity and Robustness in Predicting Yields: A Benchmarking Study of Amide Coupling Reactions. *chemrxiv.org*, 2023. 1

[8] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*, 2017. 1

[9] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural Message Passing for Quantum Chemistry. In *ICML*, pages 1263–1272, 2017. 1

[10] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation Learning on Graphs with Jumping Knowledge Networks. In *ICML*, pages 5453–5462, 2018. 1

[11] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. In *ICLR*, 2018. 1

[12] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A Continuous-Filter Convolutional Neural Network for Modeling Quantum Interactions. In *NIPS*, pages 991–1001, 2017. 1, 5

[13] Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional Message Passing for Molecular Graphs. In *ICLR*, 2020. 1, 5

[14] Johannes Gasteiger, Florian Becker, and Stephan Günnemann. GemNet: Universal Directional Graph Neural Networks for Molecules. In *NeurIPS*, pages 6790–6802, 2021. 1, 5

[15] Kristof Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant Message Passing for the Prediction of Tensorial Properties and Molecular Spectra. In *ICML*, pages 9377–9388, 2021. 1, 5

[16] Weitao Du, He Zhang, Yuanqi Du, Qi Meng, Wei Chen, Nanning Zheng, Bin Shao, and Tie-Yan Liu. SE(3) Equivariant Graph Neural Networks with Complete Local Frames. In *ICML*, pages 5583–5608, 2022. 1, 5

[17] Weitao Du, Yuanqi Du, Limei Wang, Dieqiao Feng, Guifeng Wang, Shuiwang Ji, Carla Gomes, and Zhi-Ming Ma. A New Perspective on Building Efficient and Expressive 3D Equivariant Graph Neural Networks. *arXiv.org*, 2023. 1, 5

[18] Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and Leman Go Neural: Higher-Order Graph Neural Networks. In *AAAI*, pages 4602–4609, 2019. 1

[19] Keir Adams, Lagnajit Pattanaik, and Connor W. Coley. Learning 3D Representations of Molecular Chirality with Invariance to Bond Rotations. In *ICLR*, 2022. 1, 6

[20] Bharath Ramsundar, Peter Eastman, Patrick Walters, and Vijay Pande. *Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery, and More*. O'Reilly Media, 2019. 2

[21] Emanuele Perola and Paul S. Charifson. Conformational Analysis of Drug-Like Molecules Bound to Proteins: An Extensive Study of Ligand Reorganization upon Binding. *J. Med. Chem.*, 47(10):2499–2510, 2004. 2, 3

[22] Andrew F. Zahrt, Jeremy J. Henle, Brennan T. Rose, Yang Wang, William T. Darrow, and Scott E. Denmark. Prediction of Higher-Selectivity Catalysts by Computer-Driven Workflow and Machine Learning. *Science*, 363(6424):eaau5631, 2019. 2

[23] Simon Axelrod and Rafael Gómez-Bombarelli. Molecular Machine Learning with Conformer Ensembles. *arXiv.org*, 2020. 2, 7

[24] Jan Weinreich, Nicholas J. Browning, and O. Anatole von Lilienfeld. Machine Learning of Free Energies in Chemical Compound Space Using Ensemble Representations: Reaching Experimental Uncertainty for Solvation. *J. Chem. Phys.*, 154(13):134113, 2021. 2

[25] Kangway V. Chuang and Michael J. Keiser. Attention-Based Learning on Molecular Ensembles. *arXiv.org*, 2020. 2

[26] Chaitanya K. Joshi, Arian R. Jamasb, Ramon Viñas, Charles Harris, Simon Mathis, and Pietro Liò. Multi-State RNA Design with Geometric Multi-Graph Neural Networks. *arXiv.org*, 2023. 2

[27] Simon Axelrod and Rafael Gómez-Bombarelli. GEOM, Energy-Annotated Molecular Conformations for Property Prediction and Molecular Generation. *Sci. Data*, 9(1):185, 2022. 3, 14

[28] Zhen Liu, Tetiana Zubatiuk, Adrian Roitberg, and Olexandr Isayev. Auto3D: Automatic Generation of the Low-Energy 3D Structures with ANI Neural Network Potentials. *J. Chem. Inf. Model.*, 62(22):5373–5382, 2022. 3, 14

[29] Roman Zubatyuk, Justin S. Smith, Benjamin T. Nebgen, Sergei Tretiak, and Olexandr Isayev. Teaching a Neural Network to Attach and Detach Electrons From Molecules. *Nat. Commun.*, 12(1):1–11, 2021. 3, 14

[30] Andrzej M. Żurański, Jason Y. Wang, Benjamin J. Shields, and Abigail G. Doyle. Auto-QChem: An Automated Workflow for the Generation and Storage of DFT Calculations for Organic Molecules. *React. Chem. Eng.*, 7(6):1276–1284, 2022. 3, 14

[31] Qiyuan Zhao, Sai Mahit Vaddadi, Michael Woulfe, Lawal A. Ogunfowora, Sanjay S. Garimella, Olexandr Isayev, and Brett M. Savoie. Comprehensive Exploration of Graphically Defined Reaction Spaces. *Sci. Data*, 10(1):1–10, 2023. 3, 14

[32] Peikun Zheng, Roman Zubatyuk, Wei Wu, Olexandr Isayev, and Pavlo O. Dral. Artificial Intelligence-Enhanced Quantum Chemical Method with Broad Applicability. *Nat. Commun.*, 12(1):7022, 2021. 3

[33] Tobias Gensch, Gabriel dos Passos Gomes, Pascal Friederich, Ellyn Peters, Théophile Gaudin, Robert Pollice, Kjell Jorner, AkshatKumar Nigam, Michael Lindner-D'Addario, Matthew S. Sigman, and Alán Aspuru-Guzik. A Comprehensive Discovery Platform for Organophosphorus Ligands for Catalysis. *J. Am. Chem. Soc.*, 144:1205–1217, 2022. 3, 15

[34] Anthony R. Rosales, Jessica Wahlers, Elaine Limé, Rebecca E. Meadows, Kevin W. Leslie, Rhona Savin, Fiona Bell, Eric Hansen, Paul Helquist, Rachel H. Munday, Olaf Wiest, and Per-Ola Norrby. Rapid Virtual Screening of Enantioselective Catalysts Using CatVS. *Nat. Catal.*, 2(1):41–45, 2019. 4, 15, 16

[35] G. P. Moss. Basic Terminology of Stereochemistry (IUPAC Recommendations 1996). *Pure Appl. Chem.*, 68(12):2193–2222, 1996. 4, 16

[36] Benjamin Meyer, Boodsarin Sawatlon, Stefan Heinen, O. Anatole von Lilienfeld, and Clémence Corminboeuf. Machine Learning Meets Volcano Plots: Computational Discovery of Cross-Coupling Catalysts. *Chem. Sci.*, 9:7069–7077, 2018. 4, 16

[37] Noel M. O'Boyle, Michael Banck, Craig A. James, Chris Morley, Tim Vandermeersch, and Geoffrey R. Hutchison. Open Babel: An Open Chemical Toolbox. *J. Cheminformatics*, 3(1):1–14, 2011. 4, 16

[38] Greg Landrum, Paolo Tosco, Brian Kelley, Ric, sriniker, gedeck, Riccardo Vianello, NadineSchneider, Eisuke Kawashima, Andrew Dalke, Dan N, David Cosgrove, Brian Cole, Matt Swain, Samo Turk, AlexanderSavelyev, Gareth Jones, Alain Vaucher, Maciej Wójcikowski, Ichiru Take, Daniel Probst, Kazuya Ujihara, Vincent F. Scalfani, guillaume godin, Axel Pahl, Francois Berenger, JLVarjo, strets123, JP, and DoliathGavid. rdkit/rdkit: 2022_03_2 (q1 2022) release, 2022. 4, 5, 16

[39] Darko Butina. Unsupervised Data Base Clustering Based on Daylight's Fingerprint and Tanimoto Similarity: A Fast and Automated Way To Cluster Small and Large Data Sets. *J. Chem. Inf. Comput. Sci.*, 39(4): 747–750, 1999. 4

[40] Leo Breiman. Random Forests. *Mach. Learn.*, 45(1):5–32, 2001. 5

[41] David Rogers and Mathew Hahn. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.*, 50:742–754, 2010. 5, 16

[42] Joseph L. Durant, Burton A. Leland, Douglas R. Henry, and James G. Nourse. Reoptimization of MDL Keys for Use in Drug Discovery. *J. Chem. Inf. Comput. Sci.*, 42(6):1273–1280, 2002. 5, 16

[43] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Comp.*, 9(8):1735–1780, 1997. 5

[44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Uszkoreit Kaiser, and Illia Polosukhin. Attention is All You Need. In *NIPS*, pages 5998–6008, 2017. 5, 16

[45] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful Are Graph Neural Networks? In *ICLR*, 2019. 5

[46] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open Graph Benchmark: Datasets for Machine Learning on Graphs. In *NeurIPS*, pages 22118–22133, 2020. 5, 17

[47] Kevin Yang, Kyle Swanson, Wengong Jin, Connor W. Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, Andrew Palmer, Volker Settels, Tommi S. Jaakkola, Klavs F. Jensen, and Regina Barzilay. Analyzing Learned Molecular Representations for Property Prediction. *J. Chem. Inf. Model.*, 59(8):3370–3388, 2019. 5

[48] Ladislav Rampášek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a General, Powerful, Scalable Graph Transformer. In *NeurIPS*, pages 14501–14515, 2022. 5

[49] Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) Equivariant Graph Neural Networks. In *ICML*, pages 9323–9332, 2021. 5

[50] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the Multiple Instance Problem with Axis-Parallel Rectangles. *Artif. Intell.*, 89(1-2):31–71, 1997. 6

[51] Oded Maron and Tomás Lozano-Pérez. A Framework for Multiple-Instance Learning. In *NIPS*, pages 570–576, 1997. 6

[52] Maximilian Ilse, Jakub M. Tomczak, and Max Welling. Attention-based Deep Multiple Instance Learning. In *ICML*, pages 2132–2141, 2018. 6

[53] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan R. Salakhutdinov, and Alexander J. Smola. Deep Sets. In *NIPS*, pages 3391–3401, 2017. 6

[54] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In *ICLR*, 2015. 6

[55] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 2015. 7, 17

[56] Carlo Adamo and Vincenzo Barone. Toward Reliable Density Functional Methods Without Adjustable Parameters: The PBE0 Model. *J. Chem. Phys.*, 110(13):6158–6170, 1999. 14

[57] A. Verloop, W. Hoogenstraaten, and J. Tipker. Development and Application of New Steric Substituent Parameters in Drug Design. In E. J. Ariëns, editor, *Drug Design*, volume 11 of *Medicinal Chemistry: A Series of Monographs*, pages 165–207. Academic Press, Amsterdam, 1976. 15

[58] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011. 16

[59] Philip Gage. A New Algorithm for Data Compression. *C Users J.*, 12(2):23–38, 1994. 16

[60] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*, pages 8024–8035, 2019. 17

[61] Matthias Fey and Jan Eric Lenssen. Fast Graph Representation Learning with PyTorch Geometric. In *RLGM@ICLR*, 2019. 17

Supplementary Material for MARCEL

A Dataset Description **14**
A.1 Drugs-75K 14
A.2 Kraken 15
A.3 EE 15
A.4 BDE 16

B Implementation Details **16**
B.1 Implementation of 1D Models 16
B.2 Featurizations of Molecules for 2D Models 17
B.3 Hyperparameter Specifications and Experimental Environments 17

C Additional Experiments on Evaluation Schemes of the Conformer Sampling Strategy **17**

D Raw Data **18**

A Dataset Description

MARCEL include four datasets that cover a diverse range of chemical space, which focuses on four chemically-relevant tasks for both molecules and reactions, with an emphasis on Boltzmann-averaged properties of conformer ensembles computed at the Density-Functional Theory (DFT) level. Detailed information regarding dataset access, data formatting, and loading procedures can be found at our GitHub repository `https://github.com/SXKDZ/MARCEL`. Any subsequent updates will also be posted on this repository.

A.1 Drugs-75K

Drugs-75K is a subset of the GEOM-Drugs [27] dataset, which includes 75,099 drug-like molecules with at least 5 rotatable bonds. The original GEOM-Drugs dataset was constructed using semi-empirical DFT methods, which is less accurate than full DFT. To curate the Drugs-75K subset, Auto3D [28] is used to generate and optimize the conformer ensembles for each molecule and AIMNet-NSE [29] is used to calculate three important DFT-based reactivity descriptors: ionization potential, electron affinity, and electronegativity [30].

Auto3D [28] efficiently generates high-quality conformers, with a mean RMSD at around 0.2 Å when compared with DFT conformers. It has been used in other large conformer dataset generation [31]. Regarding the neural network surrogate AIMNET-NSE [29], it mimics the PBE0/ma-def2-SVP method of DFT, which is widely used in the chemistry community. Investigating their accuracy is out of the scope of this paper, but are readily accessible from multiple sources [29, 56].

Objectives. The tasks are to predict the Boltzmann-averaged value of each property across the conformer ensemble $\langle y \rangle_{k_B} = \sum_{\boldsymbol{C}_i \in \mathcal{C}} p_{\boldsymbol{C}_i} y_{\boldsymbol{C}_i}$, where $y_{\boldsymbol{C}_i}$ is a conformer-specific property. We are given each $\boldsymbol{C}_i$, and the goal is to predict $\langle y \rangle_{k_B}$ from the molecular graph G, a single conformer $\boldsymbol{C}_i \in \mathcal{C}$, or the set $\mathcal{C}$.

Dataset preparation. In preparing the 75K version of GEOM-Drugs, we begin with the original SMILES strings of the molecules. We first exclude molecules that have less than 5 rotatable bonds. To enable the utilization of AIMNet-NSE for descriptor computation, we retain only those molecules containing atoms of H, C, N, O, F, Si, P, S, and Cl. Further, we generate DFT-level conformers and compute their energies with Auto3D. Based on these conformers, we compute three chemical bond energy descriptors using AIMNet-NSE. We exclude conformers that Auto3D fails to converge and charged molecules that are unable to be processed by AIMNet-NSE, which results in 75,099

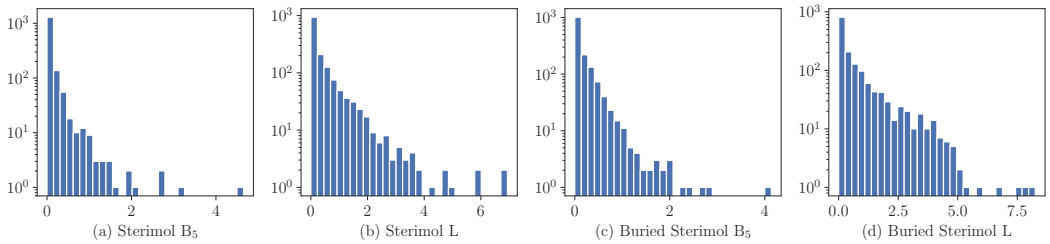

Figure S1: Histogram of the ratio of the variance of each conformer property to the variance of each Boltzmann-averaged property in the Kraken dataset.

molecules. Subsequently, we compute molecular-level Boltzmann-averaged descriptors based on conformer-level descriptors. Finally, we undertake a deduplication process as outlined in Section 3 with a RMSD threshold of 2.0, which yields a total of 558,002 distinct conformers.

Data availability and license. The original GEOM-Drugs dataset is publicly available at https://github.com/learningmatter-mit/geom but no license is specified. Our Drugs-75K can be accessed at https://github.com/SXKDZ/MARCEL/tree/main/datasets/Drugs. As for the conformer ensembles and descriptors that we generated, they are licensed under the Apache License.

A.2 KRAKEN

Kraken [33] is a dataset of 1,552 monodentate organophosphorus (III) ligands along with their DFT-computed conformer ensembles. In this study, we consider four 3D catalytic ligand descriptors exhibiting significant variance among conformers: Sterimol B_5, Sterimol L, buried Sterimol B_5, and buried Sterimol L. These descriptors quantify the steric size of a substituent in Å, and are commonly employed for Quantitative Structure-Activity Relationship (QSAR) modeling. The buried Sterimol variants describe the steric effects within the first coordination sphere of a metal [57].

Objectives. As in the Drugs-75K tasks, the goal is to predict the Boltzmann-averaged value of each property across the conformer ensemble from the molecular graph G, a single conformer $\boldsymbol{C}_i \in \mathcal{C}$, or the set $\mathcal{C}$.

Dataset preparation. In this study, we utilize the original 3D geometry structures of molecules and their corresponding Boltzmann-averaged properties provided in the Kraken dataset. Among the 78 physical-organic properties listed in the original dataset, we select four properties that demonstrate high variance across conformer ensembles, as illustrated in Figure S1.

Data availability and license. The Kraken dataset is publicly accessible at https://kraken.cs.toronto.edu. Its copyright is retained by the original authors. Under the permission of the original authors, the Kraken dataset with the conformer ensembles and the four conformer-level descriptors used in this study can be accessed at https://github.com/SXKDZ/MARCEL/tree/main/datasets/Kraken.

A.3 EE

EE [34] is a dataset of 872 catalyst-substrate pairs involving 253 Rhodium (Rh)-bound atropisomeric catalysts derived from chiral bisphosphine, with 10 enamides as substrates. The dataset includes conformations of catalyst-substrate transition state complexes in two separate pro-S and pro-R configurations. The task is to predict the Enantiomeric Excess (EE) of the chemical reaction involving the substrate, defined as the absolute ratio between the concentration of each enantiomer in the product distribution.

Objectives. EE depends on the conformer ensembles of *each* pro-R and pro-S complex. The goal is to predict EE from the graphs of the catalyst and substrate $(\mathsf{G}_{\text{cat}}, \mathsf{G}_{\text{sub}})$, a conformer $\boldsymbol{C}_i^{(\text{R})} \in \mathcal{C}^{(\text{R})}$ and $\boldsymbol{C}_i^{(\text{S})} \in \mathcal{C}^{(\text{S})}$ for each complex, or the ensembles $\mathcal{C}^{(\text{R})}$ and $\mathcal{C}^{(\text{S})}$.

Dataset preparation. The conformer ensembles are generated with Q2MM, which automatically generates Transition State Force Fields (TSFFs) in order to simulate the conformer ensembles of each

prochiral transition state complex. Then, the EE values are computed from the conformer ensembles by Boltzmann-averaging the activation energies for the competing transition states [34, 35]. Finally, we conduct the same conformer deduplication process as described in Section 3 with a RMSD threshold of 1.0.

Data availability and license. As of now, the EE dataset is proprietary, given that the publication addressing the conformer ensembles is still under preparation. Therefore, access to the EE dataset is restricted to review purposes only. We anticipate making the EE dataset publicly accessible following the acceptance of the corresponding paper.

A.4 BDE

BDE [36] is a dataset containing 5,915 organometallic catalysts ML_1L_2 consisting of a metal center (M = Pd, Pt, Au, Ag, Cu, Ni) coordinated to two flexible organic ligands (L_1 and L_2), each selected from a 91-membered ligand library. The data includes conformations of each unbound catalyst, as well as conformations of the catalyst when bound to ethylene and bromide after oxidative addition with vinyl bromide. Each catalyst has an electronic binding energy, computed as the difference in the minimum energies of the bound-catalyst complex and unbound catalyst, following the DFT-optimization of their respective conformer ensembles.

Although the binding energies are computed via DFT, the conformers provided for modeling are initially generated with Open Babel [37], followed by further geometric optimization steps, which ensures that the generated 3D structures are likely to be the global minimum energy conformers at the force field level [36, Supplementary Information]. We also note that obtaining DFT-optimized conformers for BDE is not feasible given the time-consuming nature of the process — a single geometric search using DFT can take 2 to 3 days. Therefore, this realistically represents the setting in which precise conformer ensembles are unknown at inference.

Objectives. The task is to predict the binding energy from the graphs of the unbound and bound catalyst, sampled conformers $\boldsymbol{C}_i^{(\text{unbound})} \in \mathcal{C}^{(\text{unbound})}$ and $\boldsymbol{C}_i^{(\text{bound})} \in \mathcal{C}^{(\text{bound})}$, or the ensembles $\mathcal{C}^{(\text{unbound})}$ and $\mathcal{C}^{(\text{bound})}$.

Dataset preparation. We employ Open Babel [37] to produce conformers for each unbound catalyst and each bound complex. In order to avoid redundancy, we follow a deduplication process as outlined in Section 3. For the unbound catalysts, a RMSD threshold value of 0.5 is applied, whereas for the bound complexes, a threshold of 1.0 is used.

Data availability and license. The binding energy descriptors can be accessed at https://archive.materialscloud.org/record/2018.0014/v1 under the Creative Commons Attribution 4.0 International license. The conformers are publicly available at https://github.com/SXKDZ/MARCEL/tree/main/datasets/BDE under the Apache license.

B Implementation Details

B.1 Implementation of 1D Models

For the random forest model that operates on fingerprints, we employ three molecular fingerprint schemes: the Molecular ACCess System (MACCS) [42], Extended-Connectivity Fingerprints (ECFP) [41], and RDKit topological fingerprints [38]. Then, we concatenate their outputs into a single vector, which might lead to some feature redundancy, given the possible overlaps in these three fingerprint representations of the molecular structure. To tackle this issue, we remove any features that exhibit a high correlation exceeding 90% with the other features. For implementation, we employ Scikit-Learn [58] and compute fingerprints with RDKit [38].

For both LSTM and Transformer models that operate on SMILES strings, we use a Byte-Pair Encoding (BPE)-based tokenizer [59] that is pretrained on PubChem10M, which strikes a balance among character- and word-level representations and allows to handle large vocabularies in molecular corpora. For the Transformer model, we further follow the positional embedding scheme [44] to capture the positional relationship among tokens in the SMILES string.

Table S1: A summary of node and edge features used in 2D GNN models.

	Feature	Explanation
Node	`AtomicNum`	Atomic number, representing the type of atom.
	`ChiralTag`	Indicator of chirality, a property of asymmetry.
	`TotalDegree`	Sum of implicit and explicit bonds of an atom.
	`FormalCharge`	Charge of an atom assuming equal sharing of bonding electrons.
	`TotalNumHs`	Total number of hydrogen atoms bonded to the atom.
	`NumRadicalElectrons`	Count of unpaired electrons in an atom.
	`Hybridization`	Type of atomic orbital hybridization in the atom.
	`IsAromatic`	Boolean indicating if the atom is part of an aromatic ring.
	`IsInRing`	Boolean indicating if the atom is part of any ring structure.
Edge	`BondType`	Type of the bond (e.g., single, double, triple, aromatic).
	`Stereo`	Stereochemistry of the bond (e.g., "none", "any", "Z", or "E" for double bonds).
	`IsConjugated`	Boolean indicating if the bond is part of a conjugated system.

B.2 Featurizations of Molecules for 2D Models

Following OGB [46], we employ a rich set of features for atoms (nodes) and bonds (edges) for 2D GNN models. A complete list of node and features can be found in Table S1.

B.3 Hyperparameter Specifications and Experimental Environments

Each model is trained over 2,000 epochs using the Adam optimizer [55] with early stopping triggered if there is no improvement in the training loss over 200 epochs. To ensure a fair comparison, the hidden dimension size is uniformly set to 128 for all models. Other hyperparameters mostly follow the original configurations as described in the respective papers. The complete hyperparameter set of each model can be found in `https://github.com/SXKDZ/MARCEL/tree/main/benchmarks/params`.

We utilize PyTorch [60] and PyTorch-Geometric [61] to implement all deep learning models. Most of the experiments are conducted on servers equipped with Nvidia A100 GPUs, each with 40GB of memory. For memory-intensive models such as GemNet and LEFTNet, we use servers with Nvidia H100 GPUs, each with 80GB memory. The cumulative computation time across all experiments amounts to approximately 6,000 single GPU hours.

C Additional Experiments on Evaluation Schemes of the Conformer Sampling Strategy

In this section, we conduct one additional experiment on the conformer ensemble learning strategies. We assess all 3D models on five tasks: Ionization Potential (IP) from the Drugs-75K dataset, B_5 and $BurB_5$ from the Kraken dataset, and tasks from the EE and BDE datasets.

In our previous setup, we evaluate the conformer sampling strategy using the lowest-energy conformer of each molecule at evaluation time, to provide a direct comparison to the single-conformer 3D models that are trained and tested with the lowest energy conformation. In these experiments, we continue to sample a random conformer uniformly from the conformer ensemble during training time, but consider two additional evaluation schemes: (1) evaluating model performance when encoding a randomly sampled conformer, and (2) evaluating model -performance when averaging the per-conformer predictions across the entire conformer ensemble.

The results of these experiments are summarized in Table S2. In the table, we refer to the original evaluation scheme as "fixed", and the additional schemes as "random" and "all", respectively. We find that across all three schemes, using the lowest-energy conformer for evaluation consistently yields the best performance. This is expected, as the lowest-energy conformer contributes the most to ensemble-level descriptors. The random conformer evaluation scheme generally yields the worst performance, which is likely due to the introduction of noise from less relevant conformers at test

Table S2: Performance comparison of three conformer sampling variants with different evaluation strategies. All models are trained with a randomly sampled conformer from the ensemble. The last column summarizes the average rank across all datasets for each base model.

Model	Evaluation Strategy	Drugs-75K	Kraken		EE	BDE	Average Rank
		IP	B_5	$BurB_5$			
SchNet	Fixed	0.4452	0.3235	0.2086	20.3595	1.9737	1
	Random	0.4498	0.3682	0.2454	22.0380	2.4416	3
	All	0.4428	0.3856	0.2407	18.0296	2.0106	2
DimeNet++	Fixed	0.4395	0.3323	0.2237	15.0596	1.4741	= 2
	Random	0.4555	0.3549	0.2222	13.5681	1.4688	= 2
	All	0.4479	0.3282	0.2001	12.3562	1.6270	1
GemNet	Fixed	0.4066	0.2694	0.1796	12.0541	1.6059	1
	Random	0.4250	0.4034	0.2534	16.1709	1.7894	3
	All	0.4320	0.4523	0.2481	14.3952	1.6660	2
PaiNN	Fixed	0.4466	0.3441	0.2476	19.1521	1.9262	1
	Random	0.4770	0.3756	0.2478	21.3553	1.9411	3
	All	0.4478	0.3458	0.2342	19.1955	1.8696	2
ClofNet	Fixed	0.4430	0.4524	0.2442	31.3733	2.5126	1
	Random	0.4530	0.4689	0.2736	31.3675	2.6310	= 2
	All	0.4363	0.4749	0.2855	34.3203	2.0271	= 2
LEFTNet	Fixed	0.4149	0.2834	0.2120	20.3358	1.5276	1
	Random	0.4518	0.3177	0.2344	20.3740	1.5842	3
	All	0.4274	0.3152	0.2170	18.8945	1.8663	2

time. Interestingly, we observe occasional performance improvement when averaging the predictions across all conformers in the ensemble, indicating that explicitly using ensemble-level information during evaluation can be beneficial.

D Raw Data

The raw performance data with standard deviation of Table 2 and Figure 3 is summarized in Table S3.

Table S3: Raw performance data (mean ± standard deviation) of representative 1D, 2D, 3D, and conformer ensemble MRL models in terms of absolute test error.

Category	Model		Drugs-75K			Kraken				EE	BDE
			IP	EA	χ	B_5	L	$BurB_5$	BurL		
1D	Random forest		0.4987±0.0037	0.4747±0.0022	0.2732±0.0031	0.4760±0.0041	0.4303±0.0090	0.2758±0.0180	0.1521±0.0149	61.2963±2.8640	3.0335±0.2693
	LSTM		0.4788±0.0024	0.4648±0.0002	0.2505±0.0050	0.4879±0.0280	0.5142±0.0411	0.2813±0.0041	0.1924±0.0028	64.0088±2.3708	2.8279±0.0728
	Transformer		0.6617±0.0023	0.5850±0.0031	0.4073±0.0006	0.9611±0.0813	0.8389±0.0431	0.4929±0.0369	0.2781±0.0207	62.0816±2.1789	10.0771±0.6457
2D	GIN		0.4354±0.0029	0.4169±0.0032	0.2260±0.0017	0.3128±0.0264	0.4003±0.0341	0.1719±0.0031	0.1200±0.0040	62.3065±2.9010	2.6368±0.2276
	GIN-VN		0.4361±0.0059	0.4169±0.0083	0.2267±0.0002	0.3567±0.0031	0.4344±0.0416	0.2422±0.0033	0.1741±0.0109	62.3815±2.1882	2.7417±0.2446
	ChemProp		0.4595±0.0028	0.4417±0.0045	0.2441±0.0012	0.4850±0.0068	0.5452±0.0454	0.3002±0.0086	0.1948±0.0138	61.0336±2.9715	2.6616±0.1429
	GraphGPS		0.4351±0.0049	0.4085±0.0055	0.2212±0.0054	0.3450±0.0324	0.4363±0.0133	0.2066±0.0115	0.1500±0.0138	61.6251±1.3743	2.4827±0.1992
3D	SchNet		0.4394±0.0062	0.4207±0.0021	0.2243±0.0089	0.3293±0.0068	0.5458±0.0341	0.2295±0.0111	0.1861±0.0095	17.7421±1.0899	2.5488±0.0050
	DimeNet++		0.4441±0.0087	0.4233±0.0072	0.2436±0.0075	0.3510±0.0107	0.4174±0.0397	0.2097±0.0160	0.1526±0.0072	14.6414±2.2791	1.4503±0.0370
	GemNet		0.4069±0.0007	0.3922±0.0024	0.1970±0.0039	0.2789±0.0125	0.3754±0.0086	0.1782±0.0099	0.1635±0.0063	18.0338±2.4777	1.6530±0.3081
	PaiNN		0.4505±0.0041	0.4495±0.0054	0.2324±0.0040	0.3443±0.0388	0.4471±0.0324	0.2395±0.0176	0.1673±0.0088	20.2359±1.2128	2.1261±0.0920
	ClofNet		0.4393±0.0084	0.4251±0.0066	0.2378±0.0020	0.4873±0.0093	0.6417±0.0362	0.2884±0.0166	0.2529±0.0052	33.9473±1.4633	2.6057±0.0236
	LEFTNet		0.4174±0.0007	0.3964±0.0009	0.2083±0.0054	0.3072±0.0012	0.4493±0.0261	0.2176±0.0010	0.1486±0.0095	19.7974±1.4097	1.5328±0.0567
3D +Sampling	SchNet		0.4452±0.0080	0.4232±0.0042	0.2243±0.0022	0.3235±0.0147	0.4598±0.0041	0.2086±0.0111	0.1739±0.0142	20.3595±1.5260	1.9737±0.0125
	DimeNet++		0.4395±0.0032	0.4217±0.0040	0.2432±0.0048	0.3323±0.0320	0.4153±0.0208	0.2237±0.0122	0.1561±0.0241	15.0596±0.2867	1.4741±0.0349
	GemNet		0.4066±0.0015	0.3910±0.0004	0.2027±0.0013	0.2694±0.0221	0.3488±0.0252	0.1796±0.0098	0.1184±0.0033	12.0541±0.7735	1.6059±0.1094
	PaiNN		0.4466±0.0087	0.4393±0.0045	0.2331±0.0037	0.3441±0.0161	0.4358±0.0343	0.2476±0.0070	0.1543±0.0022	19.1521±0.2386	1.9262±0.0188
	ClofNet		0.4430±0.0074	0.4237±0.0005	0.2335±0.0090	0.4524±0.0935	0.5962±0.0074	0.2442±0.0109	0.1756±0.0112	31.3733±1.9892	2.5126±0.2366
	LEFTNet		0.4149±0.0019	0.3988±0.0048	0.2141±0.0084	0.2834±0.0068	0.4407±0.0531	0.2120±0.0097	0.1547±0.0101	20.3358±0.6614	1.5276±0.0088
Ensemble	SchNet	Mean	0.4583±0.0019	0.4410±0.0018	0.2371±0.0098	0.3075±0.0151	0.4691±0.0234	0.2282±0.0206	0.1619±0.0062	20.1392±1.5748	2.5312±0.0246
		DeepSet	0.4537±0.0065	0.4396±0.0010	0.2385±0.0066	0.3105±0.0381	0.4322±0.0464	0.2249±0.0234	0.1535±0.0076	18.0495±1.2846	2.2941±0.2229
		Attention	0.4556±0.0075	0.4382±0.0125	0.2380±0.0007	0.2704±0.0187	0.4517±0.0132	0.2024±0.0183	0.1443±0.0043	14.2238±0.5451	2.6445±0.0031
	DimeNet++	Mean	0.4488±0.0086	0.4340±0.0079	0.2425±0.0060	0.2630±0.0122	0.3828±0.0331	0.1960±0.0059	0.1268±0.0060	12.0259±0.8933	1.7964±0.1260
		DeepSet	0.4126±0.0076	0.3944±0.0034	0.2267±0.0047	0.2889±0.0069	0.3468±0.0090	0.1783±0.0110	0.1339±0.0087	15.5754±2.6294	1.7533±0.0163
		Attention	0.4188±0.0024	0.4030±0.0075	0.2325±0.0028	0.3718±0.0300	0.3628±0.0259	0.1899±0.0081	0.1185±0.0105	13.3643±1.4309	2.5714±0.2149
	GemNet	Mean	0.4505±0.0052	0.4334±0.0023	0.2289±0.0032	0.2635±0.0053	0.3753±0.0036	0.1671±0.0154	0.1587±0.0029	11.6142±1.7271	2.1914±0.0605
		DeepSet	0.4187±0.0022	0.4002±0.0012	0.2169±0.0036	0.2313±0.0026	0.3386±0.0269	0.1589±0.0068	0.0947±0.0012	13.9273±1.8656	2.2532±0.2106
		Attention	0.4212±0.0017	0.4221±0.0097	0.2260±0.0056	0.2670±0.0026	0.3554±0.0147	0.1769±0.0153	0.1346±0.0075	12.0249±1.8418	2.6810±0.0223
	PaiNN	Mean	0.4591±0.0024	0.4425±0.0064	0.2360±0.0032	0.2877±0.0252	0.3950±0.0233	0.1817±0.0091	0.1472±0.0039	16.4239±0.0743	1.8744±0.1657
		DeepSet	0.4471±0.0071	0.4269±0.0033	0.2294±0.0065	0.2225±0.0218	0.3619±0.0192	0.1693±0.0111	0.1324±0.0091	13.5570±0.5505	2.2097±0.0586
		Attention	0.4641±0.0016	0.4567±0.0094	0.2471±0.0049	0.3496±0.0140	0.4109±0.0167	0.2123±0.0005	0.1506±0.0029	19.1556±2.2765	2.2335±0.1255
	ClofNet	Mean	0.4536±0.0030	0.4301±0.0007	0.2365±0.0075	0.3555±0.0193	0.4485±0.0053	0.2473±0.0076	0.2022±0.0212	19.9710±0.7745	2.0106±0.0856
		DeepSet	0.4280±0.0056	0.4033±0.0024	0.2199±0.0073	0.3228±0.0020	0.4742±0.0161	0.2263±0.0249	0.1548±0.0039	13.9647±1.2753	2.3576±0.0496
		Attention	0.4330±0.0071	0.4107±0.0048	0.2220±0.0084	0.3734±0.0267	0.4963±0.0286	0.2178±0.0186	0.1690±0.0281	26.7133±1.7225	2.6652±0.1438
	LEFTNet	Mean	0.4402±0.0062	0.4267±0.0026	0.2183±0.0007	0.2949±0.0001	0.3643±0.0352	0.2098±0.0146	0.1386±0.0007	18.9245±2.0136	2.0440±0.0076
		DeepSet	0.4167±0.0043	0.3953±0.0000	0.2069±0.0022	0.2644±0.0130	0.3866±0.0270	0.2023±0.0026	0.1441±0.0042	18.4189±1.8922	2.5165±0.3077
		Attention	0.4229±0.0059	0.4067±0.0047	0.2198±0.0011	0.3161±0.0116	0.4324±0.0292	0.2017±0.0023	0.1508±0.0075	18.9988±1.6904	2.6361±0.1560