Molformer: Motif-based Transformer on 3D Heterogeneous Molecular Graphs

Fang Wu
Columbia University
New York, USA
fw2359@columbia.edu

Dragomir Radev
Yale University
Connecticut, USA
dragomir.radev@yale.edu

Qiang Zhang
Zhejiang University
Hangzhou, China
qiang.zhang.cs@zju.edu.cn

Abstract

Procuring expressive molecular representations underpins AI-driven molecule design and scientific discovery. The research mainly focuses on atom-level homogeneous molecular graphs, ignoring the rich information in subgraphs or motifs. However, it has been widely accepted that substructures play a dominant role in identifying and determining molecular properties. To address such issues, we formulate heterogeneous molecular graphs (HMGs), and introduce Molformer to exploit both molecular motifs and 3D geometry. Precisely, we extract functional groups as motifs for small molecules and resort to the reinforcement learning to adaptively select quaternary amino acids as motifs for proteins. Then HMGs are constructed with both atom-level and motif-level nodes. To better accommodate those HMGs, we introduce a variant of Transformer named Molformer, which adopts a heterogeneous self-attention layer to distinguish the interactions between multi-level nodes. Besides, it is also coupled with a multi-scale mechanism to capture fine-grained local patterns with increasing contextual scales. An attentive farthest point sampling algorithm is also proposed to obtain the molecular representations. We validate Molformer across a few domains, including quantum chemistry, physiology, and biophysics. Experiments show that Molformer outperforms state-of-the-art baselines. Our work provides a promising way to utilize informative motifs from the perspective of multi-level graph construction.

1 Introduction

The past decade has witnessed the extraordinary success of deep learning in many scientific domains [53, 15, 16]. Inspired by these achievements, researchers have shown increasing interest in exploiting deep learning for drug discovery and material design [78] with the hope of identifying desired molecules rapidly [77]. A key aspect is how to represent molecules effectively, and graphs are a natural choice to preserve their internal structures. Therefore, Graph Neural Network (GNN) and its invariants [25, 37] have been applied to molecular representation learning with noticeable performance.

Most existing GNNs, however, only take atom-level information in homogeneous molecular graphs as input, failing to fully exploit rich semantic information in motifs. Motifs are significant subgraph patterns that frequently occur [64], and can be leveraged to uncover molecular properties [107]. For instance, a carboxyl group (COOH) acts as hydrogen-bond acceptors, contributing to better stability and higher boiling points. Besides, similar to the role of N-gram in natural language [8], molecular motifs promote the segmentation of atomic semantic meanings. While some regard motifs as additional features of atoms [61, 62], these methods increase the difficulty to separate the semantic meanings of motifs from atoms explicitly, and hinder models from viewing motifs from an integral

Preprint. Under review.

perspective. Others [34] take motifs as the only input, but ignore the influence of single atoms and infrequent substructures.

To overcome these problems, we formulate a novel heterogeneous molecular graph (HMG) comprised of both atom-level and motif-level nodes as the model input. It provides a clean interface to incorporate nodes of different levels [106] and prevents the error propagation caused by incorrect semantic segmentation of atoms. As for the determination of motifs, we adopt different strategies for different types of molecules. On the one hand, for small molecules, the motif lexicon is defined by functional groups based on domain knowledge. On the other hand, for proteins that are constituted of sequential amino acids, a reinforcement learning (RL) motif mining technique is introduced to discover the most meaningful amino acid subsequences for downstream tasks.

In order to better align with HMGs, we present Molformer, a 3D molecular representation model based on Transformer [93]. Molformer differs from preceding Transformer-based models in two major aspects. First, it distinguishes the interactions between nodes of different levels and incorporates them into the self-attention computation, named heterogeneous self-attention (HSA). Second, an Attentive Farthest Point Sampling (AFPS) algorithm is used to aggregate node features and obtain a comprehensive representation of the entire molecule.

To summarize, our contributions are as follows:

- To the best of our knowledge, we are the foremost to incorporate motifs and construct 3D heterogeneous molecular graphs for representation learning.
- We propose a novel Transformer architecture to perform on these heterogeneous molecular graphs. It has a modified self-attention to take into account the interactions between multi-level nodes, and an AFPS algorithm to integrate molecular representations.
- We empirically outperform or achieve competitive results compared to state-of-the-art baselines on several benchmarks of small molecules and proteins. Codes are available at `https://github.com/smiles724/Molformer`.

2 Preliminaries

Problem Definition. Suppose a molecule $\boldsymbol{S} = (\boldsymbol{P}, \boldsymbol{H})$ have N atoms, where $\boldsymbol{P} = \{\boldsymbol{p}_i\}_{i=1}^N \in \mathbb{R}^{N\times 3}$ describes 3D coordinates associated to each atom and $\boldsymbol{H} = \{\boldsymbol{h}_i\}_{i=1}^N \in \mathbb{R}^{N\times h}$ contains a set of h-dimension roto-translationally invariant features (e.g. atom types). $\boldsymbol{h}_i$ can be converted to a dense vector $\boldsymbol{x}_i \in \mathbb{R}^{\psi_{\text{embed}}}$ via a multi-layer perceptron (MLP). A representation learning model f acts on $\boldsymbol{S}$, obtaining its representation $\boldsymbol{r} = f(\boldsymbol{S})$. Then $\boldsymbol{r}$ is forwarded to a predictor g and attain the prediction of a biochemical property $\hat{y} = g(\boldsymbol{r})$.

Self-Attention Mechanism. Given input $\{\boldsymbol{x}_i\}_{i=1}^N$, the standard dot-product self-attention layer is:

$$\boldsymbol{q}_i = f_Q(\boldsymbol{x}_i),\ \boldsymbol{k}_i = f_K(\boldsymbol{x}_i),\ \boldsymbol{v}_i = f_V(\boldsymbol{x}_i) \tag{1}$$

$$a_{ij} = \boldsymbol{q}_i\boldsymbol{k}_j^T/\sqrt{\psi_{\text{model}}},\ \boldsymbol{z}_i = \sum_{j=1}^N \sigma(a_{ij})\boldsymbol{v}_j \tag{2}$$

where $\{f_Q, f_K, f_V\}$ are embedding transformations, and $\{\boldsymbol{q}_i, \boldsymbol{k}_i, \boldsymbol{v}_i\}$ are respectively the query, key, and value vectors with the same dimension ψ_{model}. a_{ij} is the attention that the token i pays to the token j. σ denotes the *Softmax* function and $\boldsymbol{z}_i$ is the output embedding of the token i. This formula conforms to a non-local network [97], indicating its inability to capture fine-grained patterns in a local context.

Position Encoding. Self-attention is invariant to permutation of the input [17], and position encoding (PE) is the only technique to reveal position information. PE can be based on absolute positions or relative distances [84]. The former uses raw positions and is not robust to spatial transformations. The latter manipulates the attention score by incorporating relative distances [112]: $a_{ij} = \boldsymbol{q}_i\boldsymbol{k}_j^T/\sqrt{\psi_{\text{model}}} + f_{\text{PE}}(\boldsymbol{p}_i - \boldsymbol{p}_j)$, where f_{PE} is a translation-invariant PE function. The rotation invariance can be further accomplished by taking a L2-norm $d_{ij} = ||\boldsymbol{p}_i - \boldsymbol{p}_j||_2$ [10] between the i-th and j-th atom.

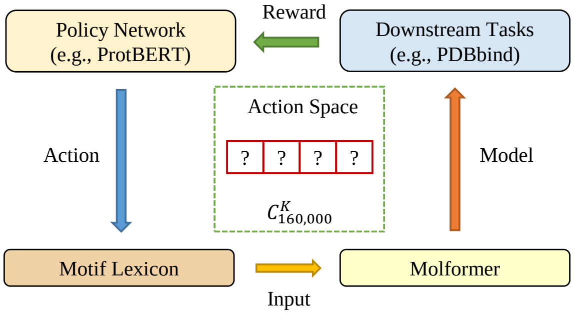


Figure 1: **The workflow of RL motif mining method in proteins.** In each iteration, the policy network is responsible for producing a motif lexicon. Molformer's performance on downstream tasks and the diversity of the lexicon are considered as the reward.

3 Heterogeneous Molecular Graphs

Motifs are frequently-occurring substructure patterns and serve as the building blocks of complex molecular structures. They have great expressiveness of the biochemical characteristics of the whole molecules [107]. We first describe how to extract motifs from small molecules and proteins respectively, and then present how to formulate HMGs.

3.1 Motifs in Small Molecules

In the chemical community, researchers have developed a set of standard criteria to recognize motifs with essential functionalities in small molecules [64]. Thus, we build motif templates of four categories of functional groups (see Figure 6, discussions are in section 6.2). Practically, we rely on RDKit [52] to draw them from SMILES [98] of small molecules.

3.2 Motifs in Proteins

In large protein molecules, motifs are local regions of 3D structures or amino acid sequences shared among proteins [6] that influence their functions [86]. Each motif usually consists of only a few elements, such as the 'helix-turn-helix' motif [43], and can describe the connectivity between secondary structural elements. The detection of protein motifs has been long studied [26]. Nevertheless, existing tools are either from the context of a protein surface [87, 86] or are task-independent and computationally expensive [7, 59]. On the basis of this peculiarity, we design an RL mining method to discover task-specific protein motifs heuristically.

Therefore, we consider motifs with four amino acids because they make up the smallest polypeptide and have special functionalities in proteins [83, 27]. For instance, β-turns are composed of four amino acids and are a non-regular secondary structure that causes a change in the direction of the polypeptide chain. Each amino acid can be of 20 different possibilities, such as Alanine, Isoleucine, and Methionine, so there are 1.6×10^5 $(= 20^4)$ potential quaternary motifs.

Our goal is to find the most effective lexicon $\boldsymbol{\mathcal{V}}^* \in \mathbb{V}$ composed of K quaternary amino acid templates, where $\mathbb{V}$ denotes the space of all $C^K_{1.6\times10^5}$ potential lexicons. Since we aim to mine the optimal task-specific lexicon, it is practically feasible to only consider the existing quaternions in the downstream datasets instead of all 1.6×10^5 possible quaternions.

In each iteration of parameter update, we use a pre-trained ProtBert [21] with a MLP as the policy network π_θ. Specifically, all possible quaternions are fed in to ProtBert to obtain their corresponding representations $\{\boldsymbol{e}_i\}_{i=1}^{1.6\times10^5}$, which are subsequently sent to the MLP to acquire their scores $\{s_i\}_{i=1}^{1.6\times10^5}$. These scores illustrates each quaternion's significance to benefit the downstream tasks if they are chosen as a part of the vocabulary. Then top-K motifs with the highest scores are selected to comprise $\boldsymbol{\mathcal{V}} \in \mathbb{V}$ in accordance with $\{s_i\}_{i=1}^{1.6\times10^5}$, and $\boldsymbol{\mathcal{V}}$ is used as templates to extract motifs and construct HMGs in downstream tasks. After that, a Molformer is trained based on these HMGs. Its validation performance is regarded as the reward r to update parameters θ by means of policy gradients [89]. With adequate iterations, the agent can select the optimal task-specific quaternary motif lexicon $\boldsymbol{\mathcal{V}}^*$.

Remarkably, our motif mining process is a one-step game, since the policy network π_θ only generates the vocabulary $\boldsymbol{\mathcal{V}}$ once in each iteration. Thus, the trajectory consists of only one action, and the performance of Molformer based on the chosen lexicon $\boldsymbol{\mathcal{V}}$ composes a part of the total reward. Moreover, we also consider the diversity of motif templates within the lexicon, and calculate it as:

$$d_{\text{div}}(\boldsymbol{\mathcal{V}}) = \frac{1}{|\boldsymbol{\mathcal{V}}|}\sum_{\boldsymbol{m}_i\in\boldsymbol{\mathcal{V}}}\sum_{\boldsymbol{m}_j\in\boldsymbol{\mathcal{V}}} d_{\text{lev}}(\boldsymbol{m}_i,\boldsymbol{m}_j) \tag{3}$$

where d_{lev} is the Levenshtein distance of two quaternary sequences $\boldsymbol{m}_i$ and $\boldsymbol{m}_j$. The final reward therefore becomes $R(\boldsymbol{\mathcal{V}}) = r + \gamma d_{\text{div}}(\boldsymbol{\mathcal{V}})$, where γ is a weight to balance two reward terms, and the policy gradient is computed as the following objective:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\boldsymbol{\mathcal{V}}\in\mathbb{V}}[\nabla_\theta \log \pi_\theta(\boldsymbol{\mathcal{V}})R(\boldsymbol{\mathcal{V}})] \tag{4}$$

3.3 Formulation of Heterogeneous Molecular Graphs

Most prior studies [61, 77, 62] simply incorporate motifs into atom features. For instance, they differentiate carbons into aromatic or non-aromatic and deem it as extra features. We argue its ineffectiveness for two reasons. First, the fusion of multi-level features increases the difficulty to summarize the functionality of motifs. Second, it hinders models to see motifs from a unitary perspective. To fill these gaps, we separate apart motifs and atoms, regarding motifs as new nodes to formulate a HMG. This way disentangles motif-level and atom-level representations, thus alleviating the difficulty for models to properly mine the motif-level semantic meanings.

Similar to the relation between phrases and single words in natural language, motifs in molecules carry higher-level semantic meanings than atoms. Therefore, they plays an essential part in identifying the functionalities of their atomic constituents. Inspired by the employment of dynamic lattice structures in named entity recognition [55], we treat each category of motifs as a new type of node and build HMGs as the input of our Molformer. To begin with, motifs are extracted according to a motif vocabulary $\boldsymbol{\mathcal{V}}$. We assume M motifs $\{\boldsymbol{m}_i\}_{i=1}^{M}$ are detected in the molecule $\boldsymbol{S}$. Consequently, a HMG includes both the motif-level and atom-level nodes as $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_N, \boldsymbol{x}_{\boldsymbol{m}_1}, ..., \boldsymbol{x}_{\boldsymbol{m}_M}\}$, where $\boldsymbol{x}_{\boldsymbol{m}_i} \in \mathbb{R}^{\psi_{\text{embed}}}$ is obtained through a learnable embedding matrix $\boldsymbol{W}^M \in \mathbb{R}^{C' \times \psi_{\text{embed}}}$ and C' denotes the number of motif categories. As for positions of each motif, we adopt a weighted sum of 3D coordinates of its components as $\boldsymbol{p}_{\boldsymbol{m}_i} = \sum_{\boldsymbol{x}_i\in\boldsymbol{m}_i}\left(\frac{w_i}{\sum_{\boldsymbol{x}_i\in\boldsymbol{m}_i} w_i}\right)\cdot \boldsymbol{p}_i$, where w_i are the atomic weights. Analogous to word segmentation [110], our HMGs composed of multi-level nodes avoid the error propagation due to inappropriate semantic segmentation while leveraging the atom information for molecular representation learning.

4 Molformer

Molformer modifies Transformer with several novel components specifically designed for 3D HMGs. First, we build a motif vocabulary and match each molecule with this lexicon to obtain all its contained motifs. Then both atoms and motifs acquire their corresponding embeddings and are forwarded into L feature learning blocks. Each block consists of a HSA, a feed-forward network (FFN) and two layer normalizations. After that, an AFPS is followed to adaptively produce the molecular representation, which is later fed into a dense predictor to forecast properties in a broad range of downstream tasks.

4.1 Heterogeneous Self-attention

After formulating a HMG with N atom-level nodes and M motif-level nodes, it is important to endow the model with the capacity of separating the interactions between multi-order nodes. To this end, we utilize a function $\phi(i,j) : \mathbb{R}^{(N+M)\times(N+M)} \rightarrow \mathbb{Z}$, which identifies the relation between any two nodes into three sorts: atom-atom, atom-motif, and motif-motif. Then a learnable scalar $b_{\phi(i,j)} : \mathbb{Z} \rightarrow \mathbb{R}$ indexed by $\phi(i,j)$ is introduced so that each node can adaptively attend to all other nodes according to their hierarchical relationship inside our HMGs [105].

In addition, we consider to exploit 3D molecular geometry (see Figure 2). Since robustness to global changes such as 3D translations and rotations is an underlying principle for molecular representation learning, we seek to satisfy roto-translation invariance. There, we borrow ideas from

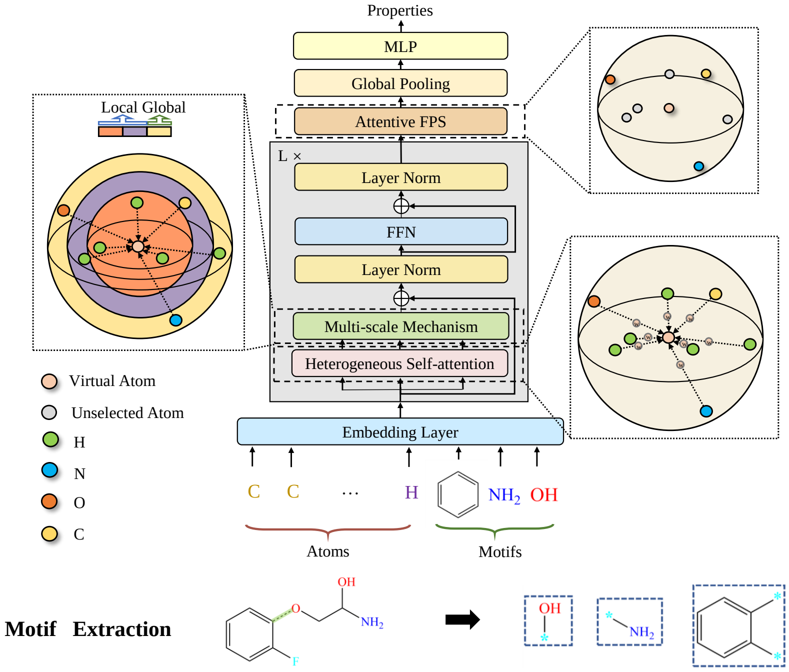


Figure 2: **The architecture of Molformer.** Given a heterogeneous molecular graph with both atom-level and motif-level nodes, stacked feature learning blocks composed of a heterogeneous self-attention module and a FFN compute their updated features. Afterwards, an attentive subsampling module integrates the molecular representation for downstream predictions. Local features with different scales are in purple and orange; yellow corresponds to global features.

SE(3)-Transformer [82] and AlphaFold2 [45], and apply a convolutional operation to the pairwise distance matrix $\boldsymbol{D} = [d_{i,j}]_{i,j\in[N+M]} \in \mathbb{R}^{(N+M)\times(N+M)}$ as $\hat{\boldsymbol{D}} = \text{Conv}_{2d}(\boldsymbol{D})$, where Conv_{2d} denotes a 2D shallow convolutional network with a kernel size of 1×1. Consequently, the attention score is computed as follows:

$$\hat{a}_{ij} = \left(\boldsymbol{q}_i \boldsymbol{k}_j^T / \sqrt{\psi_{\text{model}}}\right) \cdot \hat{d}_{ij} + b_{\phi(i,j)}, \tag{5}$$

where $\hat{d}_{i,j} \in \hat{\boldsymbol{D}}$ controls the impact of interatomic distance over the attention score, and $b_{\phi(i,j)}$ is shared across all layers.

Moreover, exploiting local context has proven to be important in sparse 3D space [71]. However, it has been pointed out that self-attention is good at capturing global data patterns but ignores local context [28, 100]. Based on this fact, we impose a distance-based constraint in self-attention in order to extract multi-scaled patterns from both local and global contexts. Guo et al. [29] propose to use integer-based distances to limit attention to local word neighbors, which cannot be used in molecules. This is because different types of molecules have different densities and molecules of the same type have different spatial regularity, which results in the non-uniformity of interatomic distances. Normally, small molecules have a mean interatomic distance of 1-2 Å (Angstrom, $10^{-10}m$), which is denser than large molecules like proteins with approximately 5 Å on average. To tackle that, we design a multi-scale methodology to robustly capture details. Specifically, we mask nodes beyond a certain distance τ_s (a real number as opposed to an integer in [29]) at each scale s. The attention calculation is modified as:

$$a_{ij}^{\tau_s} = \hat{a}_{ij} \cdot \mathbb{1}_{\{d_{ij}<\tau_s\}}, \boldsymbol{z}_i^{\tau_s} = \sum_{j=1}^{N} \sigma\left(a_{ij}^{\tau_s}\right) \boldsymbol{v}_j, \tag{6}$$

where $\mathbb{1}_{\{d_{ij}<\tau_s\}}$ is the indicator function. Notably, Equation 6 can be complementarily combined with Equation 5. Then features extracted from S different scales $\{\tau_s\}_{s=1}^{S}$ as well as the informative global feature are concatenated together to form a multi-scale representation, denoted by $\boldsymbol{z}_i' = \boldsymbol{z}_i^{\tau_1} \oplus \ldots \oplus \boldsymbol{z}_i^{\tau_S} \oplus \boldsymbol{z}_i^{global} \in \mathbb{R}^{(S+1)\psi_{\text{model}}}$. After that, $\boldsymbol{z}_i'$ is forwarded into a FFN to obtain $\boldsymbol{z}_i''$ with the original dimension ψ_{model}.

Algorithm 1 Attentive Farthest Point Sampling

```
Input: The attention score matrix $\boldsymbol{A} \in \mathbb{R}^{(N+M)\times(N+M)}$, a Euclidean distance matrix $\boldsymbol{D} \in \mathbb{R}^{(N+M)\times(N+M)}$.
Output: $K$ sampled points.
$\tilde{\boldsymbol{A}} \leftarrow \sum_i \hat{a}_{ij} \in \mathbb{R}^{N+M}$                                        ▷ sum up along rows
$\tilde{\boldsymbol{D}} \leftarrow \frac{\boldsymbol{D}-\min \boldsymbol{D}}{\max \boldsymbol{D}-\min \boldsymbol{D}} \in \mathbb{R}^{(N+M)\times(N+M)}$            ▷ normalize the distance matrix
$\mathcal{P} = \{\boldsymbol{x}_\#\}$
$\mathcal{M} = \{\boldsymbol{x}_i\}_{i=1}^{N+M}$
while length$(\mathcal{P}) < k$ do
    $\boldsymbol{x}_{\text{new}} \leftarrow \underset{i\in\mathcal{M}}{\text{argmax}}\,(\min_{j\in\mathcal{P}} \tilde{D}_{ij} + \lambda \tilde{A}_i)$        ▷ pick up the node that maximize the objective
    $\mathcal{P}$.append($\boldsymbol{x}_{\text{new}}$)
    $\mathcal{M}$.remove($\boldsymbol{x}_{\text{new}}$)
end whilereturn $\mathcal{P}$
```

4.2 Attentive Farthest Point Sampling

After having the node embeddings $\{\boldsymbol{z}_i''\}_{i=1}^{N+M}$, we study how to obtain the molecular representation $\boldsymbol{r}$. For GNNs, several readout functions such as set2set [95] and GG-NN [25] are invented. For Transformers, one way is via a virtual node. Though Ying et al. [105] state that it significantly improves the performance of existing models in the leaderboard of Open Graph Benchmark [33], this way concentrates more on its close adjacent nodes and less on distant ones, and may lead to inadvertent over-smoothing of information propagation [38]. Besides, it is difficult to locate a virtual node in 3D space and build connections to existing vertices. The other way selects a subset of nodes via a downsampling algorithm named Farthest Point Search (FPS), but it ignores nodes' differences and has sensitivity to outlier points [67] as well as uncontrollable randomness. To address these issues, we propose a new algorithm named AFPS. It aims to sample vertices by not merely spatial distances, but also their significance in terms of attention scores.

Specifically, we choose the virtual atom $\boldsymbol{x}_\#$ as the starting point and initialize two lists $\mathcal{P} = \{\boldsymbol{x}_\#\}$ and $\mathcal{M} = \{\boldsymbol{x}_i\}_{i=1}^{N+M}$ to store remaining candidate points. Then the process begins with the attention score matrix $\hat{\boldsymbol{A}} = [\hat{a}_{i,j}]_{i,j\in[N+M]} \in \mathbb{R}^{(N+M)\times(N+M)}$ and the interatomic distance matrix $\boldsymbol{D} \in \mathbb{R}^{(N+M)\times(N+M)}$. It can be easily proved that each row of $\hat{\boldsymbol{A}}$ sums up to 1 after the $Softmax$ operation along columns, i.e. $\sum_j \hat{a}_{ij} = 1, \forall i \in [N+M]$. In order to obtain the importance of each atom in the self-attention computation, we accumulate $\hat{\boldsymbol{A}}$ along rows and get $\tilde{\boldsymbol{A}} = \sum_i \hat{a}_{ij} \in \mathbb{R}^{N+M}$. Besides, we adopt the min-max normalization to rescale the distance matrix $\boldsymbol{D}$ into values between 0 and 1, and obtain $\tilde{\boldsymbol{D}} = \frac{\boldsymbol{D}-\min \boldsymbol{D}}{\max \boldsymbol{D}-\min \boldsymbol{D}}$.

After the above preprocess, we repeatedly move a point $\boldsymbol{x}_{\text{new}}$ from $\mathcal{M}$ to $\mathcal{P}$, which ensures that $\boldsymbol{x}_{\text{new}}$ is as far from $\mathcal{P}$ as possible by maximizing $\tilde{D}_{ij}$ and also plays a crucial role in attention computation by maximizing $\tilde{A}_i$. Mathematically, the AFPS aims to achieve the objective:

$$\max \sum_{i\in\mathcal{M}} \big(\min_{j\in\mathcal{P}\setminus\{i\}} \tilde{D}_{ij} + \lambda \tilde{A}_i \big), \tag{7}$$

where λ is a hyperparameter to balance those two different goals. This process is repeated until $\mathcal{P}$ has reached K points. Algorithm 1 provides a greedy approximation solution to this AFPS optimization objective for sake of computational efficiency.

After that, sampled features $\{\boldsymbol{z}_i''\}_{i\in P}$ are gathered by a Global Average Pooling layer [56] to attain the molecular representation $\boldsymbol{r} \in \mathbb{R}^{\psi_{\text{model}}}$.

Remarkably, our proposed AFPS has considerable differences and superiority over a body of previous hierarchical approaches [19, 20]. Their subsampling operations are mainly designed for protein complexities, which often have uniform structures. To be specific, they hierarchically use alpha carbons as the intermediate set of points and aggregate information at the level of those carbons for the entire complex. However, the structures of small molecules have no such a stable paradigm, and we provide a universal method to adaptively subsample atoms without any prior assumptions on the atom arrangement.

5 Experiments

We conduct extensive experiments on 7 datasets about both small molecules and large protein molecules from three different domains, including quantum chemistry, physiology, and biophysics. Appendix 5 summarises statistical information of these 7 benchmark datasets, such as the number of tasks and task types, the number of molecules and atom classes, the minimum and maximum number of atoms, and the density (mean interatomic distances) of all molecules.

Datasets. We test Molformer on a series of small molecule datasets, containing QM7 [5], QM8 [73], QM9 [72], BBBP [60], ClinTox [24], and BACE [88] [1]. QM7 is a subset of GDB-13 and composed of 7K molecules. QM8 and QM9 are subsets of GDB-17 with 22K and 133K molecule respectively. BBBP involves records of whether a compound carries the permeability property of penetrating the blood-brain barrier. ClinTox compares drugs approved through FDA and drugs eliminated due to the toxicity during clinical trials. BACE is collected for recording compounds which could act as the inhibitors of human β-secretase 1 (BACE-1).

We also inspect Molformer's ability of learning mutual relations between proteins and molecules on the PDBbind dataset [96]. Following Townshend et al. [92], we split protein-ligand complexes by protein sequence identity at 30%. As for the target, we predict $pS = -\log(S)$, where S is the binding affinity in Molar unit. In addition, we only use the pocket of each protein and put pocket-ligand pairs together as the input.

For QM9, we use the exact train/validation/test split as Townshend et al. [92]. For PDBbind, 90% of the data is used for training and the rest is divided equally between validation and test like Chen et al. [11]. For others, we adopt the scaffold splitting method with a ratio of 8:1:1 for train/validation/test as Rong et al. [77]. More implementing details can be found in Appendix A.1

Baselines For small molecules, we compare Molformer with following baselines. TF_Robust [75] takes molecular fingerprints as the input. Weave [48], MPNN [25], Schnet [81], MEGNet [11], DMPNN [104], MGCN [58], AttentiveFP [102], DimeNet [51], DimeNet++ [50], PaiNN [82], and SphereNet [57] are all graph convolutional models. Graph Transformer [9], MAT [61], R-MAT [62], SE(3)-Transformer [23], and LieTransformer [35] are Transformer-based Equivariant Neural Networks (ENNs) [91].

For PDBbind, we choose seven baselines. DeepDTA [66] and DeepAffinity [46] take in pairs of ligand and protein SMILES as input. Cormorant [3] is an ENN that represents each atom by its absolute 3D coordinates. HoloProt [86] captures higher-level fingerprint motifs on the protein surface. Schnet, 3DCNN and 3DGCN [92] are 3D methods.

6 Results and Analysis

6.1 Overall Results on Benchmark Tasks

Molecules. Table 1 and Table 2 document the overall results on small molecules datasets, where best performance is marked bold and the second best is underlined for clear comparison. It can be discovered that Molformer achieves the lowest MAE of 43.2 on QM7 and 0.009 on QM8, beating several strong baselines including DMPNN and Graph Transformer. While not completely state-of-the-art on QM9, Molformer offers competitive performance in all property regression tasks. Particularly, we outperforms all Transformer-based ENNs, including SE(3)-Transformer and LieTransformer. As for classification problems, we surpass all baselines mostly by a fairly large margin.

Table 1: For regression tasks in QM7 and QM8, lower MAE is better. For classification tasks in BBBP, ClinTox, and Bace, higher values are better.

Method	QM7	QM8	BBBP	ClinTox	BACE
TF-Robust [75]	120.6	0.024	0.860	0.765	0.824
Weave [48]	94.7	0.022	0.837	0.823	0.791
MPNN [25]	113.0	0.015	0.913	0.879	0.815
Schnet [81]	74.2	0.020	0.847	0.717	0.750
DMPNN [104]	105.8	0.014	0.919	0.897	0.852
MGCN [58]	77.6	0.022	0.850	0.634	0.734
Attentive FP [102]	126.7	0.028	0.908	0.933	0.863
Graph Transformer [9]	47.8	0.010	0.913	-	0.880
MAT [61]	102.8	-	0.728	-	0.846
R-MAT [62]	68.6	-	0.746	-	0.871
GROVE$_{\text{large}}$ [77]	89.4	0.017	0.911	0.884	0.858
Molformer	**43.2**	**0.009**	**0.926**	**0.937**	**0.884**

[1] For BBBP, ClinTox, and BACE, we use RDKit [52] to procure 3D coordinates from SMILES.

Table 2: Comparison of MAE on QM9. The methods in orange are Transformer-based methods.

Target	ϵ_{HOMO}	ϵ_{LUMO}	$\Delta\epsilon$	μ	α	R^2	ZPVE	U_0	U	H	G	c_v
Unit	eV	eV	eV	D	bohr^3	a_0^2	meV	meV	meV	meV	meV	cal/mol K
MPNN [25]	.043	.037	.069	.030	.092	.150	1.27	45	45	39	44	.800
Schnet [81]	.041	.034	.063	.033	.235	.073	1.7	14	19	14	14	.033
MEGNet $_{\text{full}}$ [11]	.038	.031	.061	.040	.083	.265	1.4	9	10	10	10	.030
MGCN [58]	.042	.057	.064	.056	**.030**	.110	**1.12**	12.9	14.4	14.6	16.2	.038
DimeNet [51]	.027	.019	.034	.028	.046	.331	1.29	8.02	7.89	8.11	8.98	.024
DimeNet++ [50]	.024	.019	.032	.029	.043	.331	1.21	6.32	6.28	6.53	7.56	.023
SphereNet [57]	**.024**	**.019**	**.032**	.026	.047	.292	1.12	6.26	7.33	6.40	8.0	**.021**
PaiNN [82]	.028	.020	.046	**.012**	.045	**.066**	1.28	**5.85**	**5.53**	**5.98**	**7.35**	.024
SE(3)-Transformer [23]	.035	.033	.053	.051	.142	–	–	–	–	–	–	–
LieTransformer [35]	.033	.029	.052	.061	.104	2.29	3.55	17	16	27	23	.041
Molformer	.025	.026	.039	.028	.041	.350	2.05	7.52	7.46	7.38	8.11	.025

Proteins. Table 3 reports the Root-Mean-Squared Deviation (RMSD), the Pearson correlation (R_p), and the Spearman correlation (R_s) on PDBbind. Molformer achieves the lowest RMSD and the best Pearson and Spearman correlations. As Wu et al. [101] claim, appropriate featurizations which hold pertinent information is significant for PDBbind. However, an important observation in our work is that deep learning approaches with the exploitation of 3D geometric information can perform better than conventional methods like DeepDTA and DeepAffinity that use a set of physicochemical descriptors but ignore 3D structures.

Table 3: Comparison of RMSD, R_p, and R_s on PDBbind.

Method	Geometry	RMSD	R_p	R_s
DeepDTA [66]	Non-3D	1.565	0.573	0.574
DeepAffinity [46]	Non-3D	1.893	0.415	0.426
Schnet [81]	3D	1.892	0.601	-
Cormorant [3]	3D	1.429	0.541	0.532
3DCNN [92]	3D	1.520	0.558	0.556
3DGCN [92]	3D	1.963	0.581	0.647
HoloProt [86]	3D	1.464	0.509	0.500
Molformer	3D	**1.386**	**0.623**	**0.651**

6.2 Ablation Study and Discussion

What Is the Effect of Each Component? We investigate the effectiveness of each component of our Molformer in Table 3. It can be observed that HSA along with HMGs substantially boosts model's performance compared with the naive method that immediately adds 3D coordinates as the atom input feature. MAE declines from 132.2 to 46.5 in QM7 while decreases from 0.0205 to 0.0097 in QM8. In addition, AFPS produces better predictions than the counterpart that utilizes the virtual node as the molecular representation (a case study of AFPS is in Appendix B.2). We also discover that the multi-scale mechanism significantly reduces RMSD from 50.1 to 46.5 on QM7, but its improvements in QM8 are much smaller. This phenomenon indicates that multi-scale mechanism is an appropriate way to alleviate the problem of inadequate training in small datasets [1]. It endows Molformer with capability to extract local features by regulating the scope of self-attention [69]. However, as the data size gets larger, Molformer does not require the assistance of multi-scale mechanism to abstract local patterns, since the parameters of convolution operators are properly trained.

Figure 3: Effects of each module on QM7, QM8 and PDBbind (RMSD).

	HSA	AFPS	HMG	QM7	QM8	PDBbind
1	-	-	-	132.2	0.0205	1.925
2	✓	-	✓	46.5	0.0097	1.441
3	✓	✓	✓	**43.2**	**0.0095**	**1.386**

What Is the Contribution of HMGs? The ideology of constructing heterogeneous graphs has already been proven successful in not only chemical knowledge graphs [22], but named entity recognition [55]. The former views the chemical characteristics obtained from domain knowledge of elements as shared nodes, while the latter converts the lattice structure into a flat structure consisting of spans. To further verify its efficacy, we compare our motif-based HMGs with the naive fusion of multi-level features. Table 4 shows a noticeable improvement of our HMGs over the other two variants.

Figure 4: Comparison of HMGs with simple feature fusion on QM7, QM8 and PDBbind (RMSD).

	QM7	QM8	PDBbind
No Motifs	132.2	0.0205	1.925
Multi-Level Fusion	89.7	0.0154	1.427
Heterogeneous Graphs	**43.2**	**0.0095**	**1.386**

Have We Found Good Candidates of Motifs? How to determine motifs is critical to HMGs. Concerning small molecules, we define motifs on the basis of functional groups, which refers to a substituent or moiety that causes molecules' characteristic chemical reactions [63, 85]. To further explore their contributions, we divide functional groups into four categories: groups that contain only carbon and hydrogen (Hydrocarbons), groups that contain halogen (Haloalkanes), groups that contain oxygen, and groups that contain nitrogen (see Appendix A.1). The ablation studies (see Figure 5) demonstrate that Molformer can gain improvements from all four groups of motifs, where Hydrocarbons and Haloalkanes are the most and the least effective types respectively. This is in line with the fact that Hydrocarbons occur most frequently in organic molecules. Moreover, the best performance is achieved when all categories are considered, implying a promising direction to discover more effective motifs. As for proteins, motifs discovered by our RL mining method share a same backbone as $\mathrm{CC(C(NC(C)C(O){=}O){=}O)NC(CNC(CN){=}O){=}O}$ (see Figure 7 in Appendix), which is a hydrogen bond donor and implies a mark to distinguish potential binding site. Moreover, the portion of those motifs in the pocket (1.38%) is nearly twice of that in other locations (0.73%), conforming to the fact that pockets are the most preferable part for ligands to bind with.

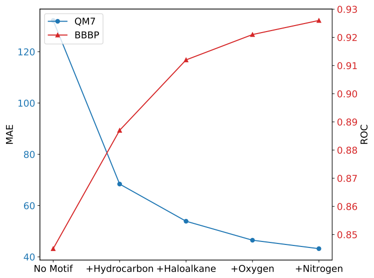


Figure 5: The ablation study of consecutively adding four motif groups (from left to right) in the QM7 and BBBP datasets.

7 Related Works

Motifs in Molecular Graphs. Motifs have been proven to benefit many tasks from exploratory analysis to transfer learning [30]. Various algorithms have been proposed to exploit them for contrastive learning [107], self-supervised pretraining [39, 77, 111], generation [42], protein design [54] and drug-drug interaction prediction [34]. To the best of our knowledge, none of them take advantage of motifs to build a heterogeneous graph for molecular property prediction.

As for motif extraction, previous motif mining methods either depend on exact counting [64, 47, 79, 12, 7] or sampling and statistical estimation [99]. No preceding studies extract task-specific motifs to enhance model performance.

Molecular Representation Learning. Deep learning has been widely applied to predict molecular properties. Molecules are usually represented as 1D linear sequences, including amino acid sequences and SMILES [40, 103], and 2D chemical bond graphs [18, 80, 104, 58]. Despite that, more evidence indicates that 3D spatial structures lead to better modelling and superior performance. 3D CNN [2, 41] and GNN [14] models have become popular to capture these complex geometries in a variety of bio-molecular applications. Nonetheless, the aforementioned methods is inefficient at grabbing local contextual feature [90] and long-range dependencies [108].

Attempts have been taken to address that issue based on the Transformer. It assumes fully-connection [94, 44] and use self-attention to capture long-term dependencies [31]. Some researchers feed SMILES in Transformer to obtain their representations [32, 70, 65, 13, 76] and conduct pretraining [13]. Some others employ Transformer to solve generative tasks [36] or fulfill equivariance [23] via spherical harmonics. However, foregoing methods are either incapable to encode 3D geometry, non-sensitive to local contextual patterns [13, 109], or inefficient to aggregate atom features [61]. More essentially, they are not specially designed to operate on heterogeneous graphs of molecules.

8 Conclusion

This paper presents Molformer for 3D molecular representation learning on heterogeneous molecular graphs. First, we extract informative motifs by means of functional groups and a reinforcement learning mining method to formulate heterogeneous molecular graphs. After that, Molformer adopts a heterogeneous self-attention to distinguish the interactions between multi-level nodes and exploit spatial information with multiplicate scales for the sake of catching local features. Then a simple but efficient downsampling algorithm is introduced to better accumulate molecular representations. Experiments show the superiority of Molformer on various domains.

References

[1] Alber, M., Tepole, A.B., Cannon, W.R., De, S., Dura-Bernal, S., Garikipati, K., Karniadakis, G., Lytton, W.W., Perdikaris, P., Petzold, L., et al., 2019. Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences. NPJ digital medicine 2, 1–11.

[2] Anand-Achim, N., Eguchi, R.R., Mathews, I.I., Perez, C.P., Derry, A., Altman, R.B., Huang, P., 2021. Protein sequence design with a learned potential. bioRxiv , 2020–01.

[3] Anderson, B., Hy, T.S., Kondor, R., 2019. Cormorant: Covariant molecular neural networks. arXiv preprint arXiv:1906.04015 .

[4] Axelrod, S., Gomez-Bombarelli, R., 2020. Geom: Energy-annotated molecular conformations for property prediction and molecular generation. arXiv preprint arXiv:2006.05531 .

[5] Blum, L.C., Reymond, J.L., 2009. 970 million druglike small molecules for virtual screening in the chemical universe database gdb-13. Journal of the American Chemical Society 131, 8732–8733.

[6] Bork, P., Koonin, E.V., 1996. Protein sequence motifs. Current opinion in structural biology 6, 366–376.

[7] Cantoni, V., Gatti, R., Lombardi, L., 2011. 3d protein surface segmentation through mathematical morphology, in: International Joint Conference on Biomedical Engineering Systems and Technologies, Springer. pp. 97–109.

[8] Cavnar, W.B., Trenkle, J.M., et al., 1994. N-gram-based text categorization, in: Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval, Citeseer.

[9] Chen, B., Barzilay, R., Jaakkola, T., 2019a. Path-augmented graph transformer network. arXiv preprint arXiv:1905.12712 .

[10] Chen, C., Li, G., Xu, R., Chen, T., Wang, M., Lin, L., 2019b. Clusternet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4994–5002.

[11] Chen, C., Ye, W., Zuo, Y., Zheng, C., Ong, S.P., 2019c. Graph networks as a universal machine learning framework for molecules and crystals. Chemistry of Materials 31, 3564–3572.

[12] Chen, J., Hsu, W., Lee, M.L., Ng, S.K., 2006. Nemofinder: Dissecting genome-wide protein-protein interactions with meso-scale network motifs, in: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 106–115.

[13] Chithrananda, S., Grand, G., Ramsundar, B., 2020. Chemberta: Large-scale self-supervised pretraining for molecular property prediction. arXiv preprint arXiv:2010.09885 .

[14] Cho, H., Choi, I.S., 2018. Three-dimensionally embedded graph convolutional network (3dgcn) for molecule interpretation. arXiv preprint arXiv:1811.09794 .

[15] Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

[16] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

[17] Dufter, P., Schmitt, M., Schütze, H., 2021. Position information in transformers: An overview. arXiv preprint arXiv:2102.11090 .

[18] Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., Adams, R.P., 2015. Convolutional networks on graphs for learning molecular fingerprints. arXiv preprint arXiv:1509.09292 .

[19] Eismann, S., Suriana, P., Jing, B., Townshend, R.J., Dror, R.O., 2020. Protein model quality assessment using rotation-equivariant, hierarchical neural networks. arXiv preprint arXiv:2011.13557 .

[20] Eismann, S., Townshend, R.J., Thomas, N., Jagota, M., Jing, B., Dror, R.O., 2021. Hierarchical, rotation-equivariant neural networks to select structural models of protein complexes. Proteins: Structure, Function, and Bioinformatics 89, 493–501.

[21] Elnaggar, A., Heinzinger, M., Dallago, C., Rihawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., et al., 2020. Prottrans: towards cracking the language of life's code through self-supervised deep learning and high performance computing. arXiv preprint arXiv:2007.06225 .

[22] Fang, Y., Yang, H., Zhuang, X., Shao, X., Fan, X., Chen, H., 2021. Knowledge-aware contrastive molecular graph learning. arXiv preprint arXiv:2103.13047 .

[23] Fuchs, F.B., Worrall, D.E., Fischer, V., Welling, M., 2020. Se (3)-transformers: 3d roto-translation equivariant attention networks. arXiv preprint arXiv:2006.10503 .

[24] Gayvert, K.M., Madhukar, N.S., Elemento, O., 2016. A data-driven approach to predicting successes and failures of clinical trials. Cell chemical biology 23, 1294–1301.

[25] Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E., 2017. Neural message passing for quantum chemistry, in: International conference on machine learning, PMLR. pp. 1263–1272.

[26] Golovin, A., Henrick, K., 2008. Msdmotif: exploring protein sites and motifs. BMC bioinformatics 9, 1–11.

[27] Gribskov, M., 2019. Identification of sequence patterns, motifs and domains .

[28] Guo, M.H., Cai, J.X., Liu, Z.N., Mu, T.J., Martin, R.R., Hu, S.M., 2020a. Pct: Point cloud transformer. arXiv preprint arXiv:2012.09688 .

[29] Guo, Q., Qiu, X., Liu, P., Xue, X., Zhang, Z., 2020b. Multi-scale self-attention for text classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 7847–7854.

[30] Henderson, K., Gallagher, B., Eliassi-Rad, T., Tong, H., Basu, S., Akoglu, L., Koutra, D., Faloutsos, C., Li, L., 2012. Rolx: structural role extraction & mining in large graphs, in: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1231–1239.

[31] Hernández, A., Amigó, J.M., 2021. Attention mechanisms and their applications to complex systems. Entropy 23, 283.

[32] Honda, S., Shi, S., Ueda, H.R., 2019. Smiles transformer: Pre-trained molecular fingerprint for low data drug discovery. arXiv preprint arXiv:1911.04738 .

[33] Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., Leskovec, J., 2020. Open graph benchmark: Datasets for machine learning on graphs. arXiv preprint arXiv:2005.00687 .

[34] Huang, K., Xiao, C., Hoang, T., Glass, L., Sun, J., 2020. Caster: Predicting drug interactions with chemical substructure representation, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 702–709.

[35] Hutchinson, M.J., Le Lan, C., Zaidi, S., Dupont, E., Teh, Y.W., Kim, H., 2021. Lietransformer: Equivariant self-attention for lie groups, in: International Conference on Machine Learning, PMLR. pp. 4533–4543.

[36] Ingraham, J., Garg, V.K., Barzilay, R., Jaakkola, T., 2019. Generative models for graph-based protein design. Advances in neural information processing systems .

[37] Ishida, S., Miyazaki, T., Sugaya, Y., Omachi, S., 2021. Graph neural networks with multiple feature extraction paths for chemical property estimation. Molecules 26, 3125.

[38] Ishiguro, K., Maeda, S.i., Koyama, M., 2019. Graph warp module: an auxiliary module for boosting the power of graph neural networks in molecular graph analysis. arXiv preprint arXiv:1902.01020 .

[39] Jaeger, S., Fulle, S., Turk, S., 2018. Mol2vec: unsupervised machine learning approach with chemical intuition. Journal of chemical information and modeling 58, 27–35.

[40] Jastrzębski, S., Leśniak, D., Czarnecki, W.M., 2016. Learning to smile (s). arXiv preprint arXiv:1602.06289 .

[41] Jiménez, J., Skalic, M., Martinez-Rosell, G., De Fabritiis, G., 2018. K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. Journal of chemical information and modeling 58, 287–296.

[42] Jin, W., Barzilay, R., Jaakkola, T., 2020. Hierarchical generation of molecular graphs using structural motifs, in: International Conference on Machine Learning, PMLR. pp. 4839–4848.

[43] Johansson, M.U., Zoete, V., Michielin, O., Guex, N., 2012. Defining and searching for structural motifs using deepview/swiss-pdbviewer. BMC bioinformatics 13, 1–11.

[44] Joshi, C., 2020. Transformers are graph neural networks. The Gradient .

[45] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al., 2021. Highly accurate protein structure prediction with alphafold. Nature 596, 583–589.

[46] Karimi, M., Wu, D., Wang, Z., Shen, Y., 2019. Deepaffinity: interpretable deep learning of compound–protein affinity through unified recurrent and convolutional neural networks. Bioinformatics 35, 3329–3338.

[47] Kashtan, N., Itzkovitz, S., Milo, R., Alon, U., 2004. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. Bioinformatics 20, 1746–1758.

[48] Kearnes, S., McCloskey, K., Berndl, M., Pande, V., Riley, P., 2016. Molecular graph convolutions: moving beyond fingerprints. Journal of computer-aided molecular design 30, 595–608.

[49] Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

[50] Klicpera, J., Giri, S., Margraf, J.T., Günnemann, S., 2020a. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. arXiv preprint arXiv:2011.14115 .

[51] Klicpera, J., Groß, J., Günnemann, S., 2020b. Directional message passing for molecular graphs. arXiv preprint arXiv:2003.03123 .

[52] Landrum, G., 2013. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling.

[53] LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. nature 521, 436–444.

[54] Li, A.J., Sundar, V., Grigoryan, G., Keating, A.E., . Terminator: A neural framework for structure-based protein design using tertiary repeating motifs .

[55] Li, X., Yan, H., Qiu, X., Huang, X., 2020. Flat: Chinese ner using flat-lattice transformer. arXiv preprint arXiv:2004.11795 .

[56] Lin, M., Chen, Q., Yan, S., 2013. Network in network. arXiv preprint arXiv:1312.4400 .

[57] Liu, Y., Wang, L., Liu, M., Zhang, X., Oztekin, B., Ji, S., 2021. Spherical message passing for 3d graph networks. arXiv preprint arXiv:2102.05013 .

[58] Lu, C., Liu, Q., Wang, C., Huang, Z., Lin, P., He, L., 2019. Molecular property prediction: A multilevel quantum interactions modeling perspective, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 1052–1060.

[59] Mackenzie, C.O., Zhou, J., Grigoryan, G., 2016. Tertiary alphabet for the observable protein structural universe. Proceedings of the National Academy of Sciences 113, E7438–E7447.

[60] Martins, I.F., Teixeira, A.L., Pinheiro, L., Falcao, A.O., 2012. A bayesian approach to in silico blood-brain barrier penetration modeling. Journal of chemical information and modeling 52, 1686–1697.

[61] Maziarka, Ł., Danel, T., Mucha, S., Rataj, K., Tabor, J., Jastrzębski, S., 2020. Molecule attention transformer. arXiv preprint arXiv:2002.08264 .

[62] Maziarka, Ł., Majchrowski, D., Danel, T., Gaiński, P., Tabor, J., Podolak, I., Morkisz, P., Jastrzębski, S., 2021. Relative molecule self-attention transformer. arXiv preprint arXiv:2110.05841 .

[63] McNaught, A.D., Wilkinson, A., et al., 1997. Compendium of chemical terminology. volume 1669. Blackwell Science Oxford.

[64] Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U., 2002. Network motifs: simple building blocks of complex networks. Science 298, 824–827.

[65] Morris, P., St. Clair, R., Hahn, W.E., Barenholtz, E., 2020. Predicting binding from screening assays with transformer network embeddings. Journal of Chemical Information and Modeling 60, 4191–4199.

[66] Öztürk, H., Özgür, A., Ozkirimli, E., 2018. Deepdta: deep drug–target binding affinity prediction. Bioinformatics 34, i821–i829.

[67] Pan, X., Xia, Z., Song, S., Li, L.E., Huang, G., 2021. 3d object detection with pointformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7463–7472.

[68] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al., 2019. Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems 32, 8026–8037.

[69] Peng, G.C., Alber, M., Tepole, A.B., Cannon, W.R., De, S., Dura-Bernal, S., Garikipati, K., Karniadakis, G., Lytton, W.W., Perdikaris, P., et al., 2021. Multiscale modeling meets machine learning: What can we learn? Archives of Computational Methods in Engineering 28, 1017–1037.

[70] Pesciullesi, G., Schwaller, P., Laino, T., Reymond, J.L., 2020. Transfer learning enables the molecular transformer to predict regio-and stereoselective reactions on carbohydrates. Nature communications 11, 1–8.

[71] Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. arXiv preprint arXiv:1706.02413 .

[72] Ramakrishnan, R., Dral, P.O., Rupp, M., von Lilienfeld, O.A., 2014. Quantum chemistry structures and properties of 134 kilo molecules. Scientific Data 1.

[73] Ramakrishnan, R., Hartmann, M., Tapavicza, E., Von Lilienfeld, O.A., 2015. Electronic spectra from tddft and machine learning in chemical space. The Journal of chemical physics 143, 084111.

[74] Ramsundar, B., Eastman, P., Walters, P., Pande, V., Leswing, K., Wu, Z., 2019. Deep Learning for the Life Sciences. O'Reilly Media. https://www.amazon.com/Deep-Learning-Life-Sciences-Microscopy/dp/1492039837.

[75] Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., Pande, V., 2015. Massively multitask networks for drug discovery. arXiv preprint arXiv:1502.02072 .

[76] Rao, R., Liu, J., Verkuil, R., Meier, J., Canny, J.F., Abbeel, P., Sercu, T., Rives, A., 2021. Msa transformer. bioRxiv .

[77] Rong, Y., Bian, Y., Xu, T., Xie, W., Wei, Y., Huang, W., Huang, J., 2020. Self-supervised graph transformer on large-scale molecular data. arXiv preprint arXiv:2007.02835 .

[78] Sanchez-Lengeling, B., Aspuru-Guzik, A., 2018. Inverse molecular design using machine learning: Generative models for matter engineering. Science 361, 360–365.

[79] Schreiber, F., Schwöbbermeyer, H., 2005. Frequency concepts and pattern detection for the analysis of motifs in networks, in: Transactions on computational systems biology III. Springer, pp. 89–104.

[80] Schütt, K.T., Arbabzadah, F., Chmiela, S., Müller, K.R., Tkatchenko, A., 2017. Quantum-chemical insights from deep tensor neural networks. Nature communications 8, 1–8.

[81] Schütt, K.T., Sauceda, H.E., Kindermans, P.J., Tkatchenko, A., Müller, K.R., 2018. Schnet–a deep learning architecture for molecules and materials. The Journal of Chemical Physics 148, 241722.

[82] Schütt, K.T., Unke, O.T., Gastegger, M., 2021. Equivariant message passing for the prediction of tensorial properties and molecular spectra. arXiv preprint arXiv:2102.03150 .

[83] Sen, D., Gilbert, W., 1988. Formation of parallel four-stranded complexes by guanine-rich motifs in dna and its implications for meiosis. nature 334, 364–366.

[84] Shaw, P., Uszkoreit, J., Vaswani, A., 2018. Self-attention with relative position representations. arXiv preprint arXiv:1803.02155 .

[85] Smith, M.B., 2020. March's advanced organic chemistry: reactions, mechanisms, and structure. John Wiley & Sons.

[86] Somnath, V.R., Bunne, C., Krause, A., 2021. Multi-scale representation learning on proteins, in: Thirty-Fifth Conference on Neural Information Processing Systems.

[87] Stepniewska-Dziubinska, M.M., Zielenkiewicz, P., Siedlecki, P., 2020. Improving detection of protein-ligand binding sites with 3d segmentation. Scientific reports 10, 1–9.

[88] Subramanian, G., Ramsundar, B., Pande, V., Denny, R.A., 2016. Computational modeling of β-secretase 1 (bace-1) inhibitors using ligand based approaches. Journal of chemical information and modeling 56, 1936–1949.

[89] Sutton, R.S., McAllester, D.A., Singh, S.P., Mansour, Y., 2000. Policy gradient methods for reinforcement learning with function approximation, in: Advances in neural information processing systems, pp. 1057–1063.

[90] Tang, G., Müller, M., Rios, A., Sennrich, R., 2018. Why self-attention? a targeted evaluation of neural machine translation architectures. arXiv preprint arXiv:1808.08946 .

[91] Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., Riley, P., 2018. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. arXiv preprint arXiv:1802.08219 .

[92] Townshend, R.J., Vögele, M., Suriana, P., Derry, A., Powers, A., Laloudakis, Y., Balachandar, S., Anderson, B., Eismann, S., Kondor, R., et al., 2020. Atom3d: Tasks on molecules in three dimensions. arXiv preprint arXiv:2012.04035 .

[93] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need, in: Advances in neural information processing systems, pp. 5998–6008.

[94] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., 2017. Graph attention networks. arXiv preprint arXiv:1710.10903 .

[95] Vinyals, O., Bengio, S., Kudlur, M., 2015. Order matters: Sequence to sequence for sets. arXiv preprint arXiv:1511.06391 .

[96] Wang, R., Fang, X., Lu, Y., Yang, C.Y., Wang, S., 2005. The pdbbind database: methodologies and updates. Journal of medicinal chemistry 48, 4111–4119.

[97] Wang, X., Girshick, R., Gupta, A., He, K., 2018. Non-local neural networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7794–7803.

[98] Weininger, D., 1988. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. Journal of chemical information and computer sciences 28, 31–36.

[99] Wernicke, S., 2006. Efficient detection of network motifs. IEEE/ACM transactions on computational biology and bioinformatics 3, 347–359.

[100] Wu, Z., Liu, Z., Lin, J., Lin, Y., Han, S., 2020. Lite transformer with long-short range attention. arXiv preprint arXiv:2004.11886 .

[101] Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V., 2018. Moleculenet: a benchmark for molecular machine learning. Chemical science 9, 513–530.

[102] Xiong, Z., Wang, D., Liu, X., Zhong, F., Wan, X., Li, X., Li, Z., Luo, X., Chen, K., Jiang, H., et al., 2019. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. Journal of medicinal chemistry 63, 8749–8760.

[103] Xu, Z., Wang, S., Zhu, F., Huang, J., 2017. Seq2seq fingerprint: An unsupervised deep molecular embedding for drug discovery, in: Proceedings of the 8th ACM international conference on bioinformatics, computational biology, and health informatics, pp. 285–294.

[104] Yang, K., Swanson, K., Jin, W., Coley, C., Eiden, P., Gao, H., Guzman-Perez, A., Hopper, T., Kelley, B., Mathea, M., et al., 2019. Analyzing learned molecular representations for property prediction. Journal of chemical information and modeling 59, 3370–3388.

[105] Ying, C., Cai, T., Luo, S., Zheng, S., Ke, G., He, D., Shen, Y., Liu, T.Y., 2021. Do transformers really perform bad for graph representation? arXiv preprint arXiv:2106.05234 .

[106] Zhang, C., Song, D., Huang, C., Swami, A., Chawla, N.V., 2019. Heterogeneous graph neural network, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 793–803.

[107] Zhang, S., Hu, Z., Subramonian, A., Sun, Y., 2020a. Motif-driven contrastive learning of graph representations. arXiv preprint arXiv:2012.12533 .

[108] Zhang, S., Liu, Y., Xie, L., 2020b. Molecular mechanics-driven graph neural network with multiplex graph for molecular structures. arXiv preprint arXiv:2011.07457 .

[109] Zhang, X.C., Wu, C.K., Yang, Z.J., Wu, Z.X., Yi, J.C., Hsieh, C.Y., Hou, T.J., Cao, D.S., 2021a. Mg-bert: leveraging unsupervised atomic representation learning for molecular property prediction. Briefings in Bioinformatics .

[110] Zhang, Y., Yang, J., 2018. Chinese ner using lattice lstm. arXiv preprint arXiv:1805.02023 .

[111] Zhang, Z., Liu, Q., Wang, H., Lu, C., Lee, C.K., 2021b. Motif-based graph self-supervised learning for molecular property prediction. arXiv preprint arXiv:2110.00987 .

[112] Zhao, H., Jiang, L., Jia, J., Torr, P., Koltun, V., 2020. Point transformer. arXiv preprint arXiv:2012.09164 .

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] Yes
   (b) Did you describe the limitations of your work? [Yes] Yes
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] 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] Yes
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Yes
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] 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] 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] Yes
   (b) Did you mention the license of the assets? [Yes] Yes
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Yes
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] Yes
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] Yes
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]

A Experimental Setup

A.1 Experimental Details

Molformer Architecture. A standard Molformer has 6 heterogeneous self-attention layers, and each layer has 3 scales and 8 heads. Normally, scales are set by $\tau = [\frac{\rho}{2}, \rho, 2\rho]$, where ρ is the density of each corresponding dataset. The number of selected atoms K and the weight ratio λ in AFPS is set as 4 and 0.1, respectively. We use ReLU as the activation function and a dropout rate of 0.1 for all layers if not specified. The input embedding size is 512 and the hidden size for FFN is 2048. For BBBP and ClinTox, we use Molformer with 2 heterogeneous self-attention layers with 4 heads. The scales are 0.8, 1.6, and 3.0 Å. The dropout rate is 0.2 and 0.6 for BBBP and ClinTox, respectively. For BACE, we use a standard Molformer but with a dropout rate of 0.2.

Training Details. We use Pytorch [68] to implement Molformer and data parallelism in two GeForce RTX 3090. An Adam [49] optimizer is used and a ReduceLROnPlateau scheduler is enforced to adjust it with a factor of 0.6 and a patience of 10. We apply no weight decay there. Each model is trained with 300 epochs, except for PDBbind where we solely train the model for 30 epochs.

For QM7 and QM8, we use a batch size of 128 and a learning rate of 10^{-4}. For QM9, we use a batch size of 256 and a learning rate of 10^{-3}. For PDBbind, we use a batch size of 16 and a learning rate of 10^{-4}. All hyper-parameters are tuned based on validation sets. For all molecular datasets, we impose no limitation on the input length and normalise the values of each regression task by mean and the standard deviation of the training set. We used grid search to tune the hyper-parameters of our model and baselines based on the validation dataset (see Table 4).

Table 4: The training hyper-parameters.

Hyper-parameter	Description	Range
bs	The input batch size.	[16, 128, 256]
lr	The initial learning rate of ReduceLROnPlateau learning rate scheduler.	$[1e-4, 1e-5]$
min_lr	The minimum learning rate of ReduceLROnPlateau learning rate scheduler.	$5e-7$
dropout	The dropout ratio.	[0.1, 0.2, 0.6]
n_encoder	The number of heterogeneous self-attention layers.	[2, 6]
head	The number of self-attention heads.	[4, 8]
embed_dim	The dimension of input embeddings.	512
ffn_dim	The hidden size of MLP layers.	1024
k	The number of sampled points in AFPS.	[4, 8, 10, 20]
lambda	The balance ratio in AFPS.	[0.1, 0.5, 1.0, 2.0]
dist_bar	The scales of the multi-scale mechanism (Å).	[[0.75, 1.55, 3.0], [0.8, 1.6, 3.0], [1.5, 3.0, 6.0]]

Motif Extraction. We adopt RDKit [52] to search motifs from SMILES representations of small molecules. For QM9 and PDBbind, Atom3D [92] provides both 3D coordinates and SMILES. For QM7, DeepChem [74] offers SMILES. For QM8, we first attain SMILES from their 3D representations using RDKit and then extract motifs.

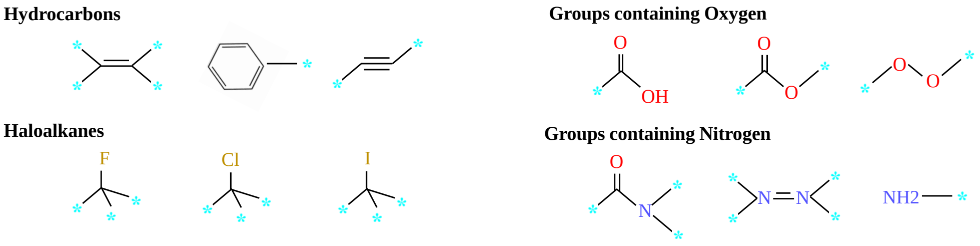
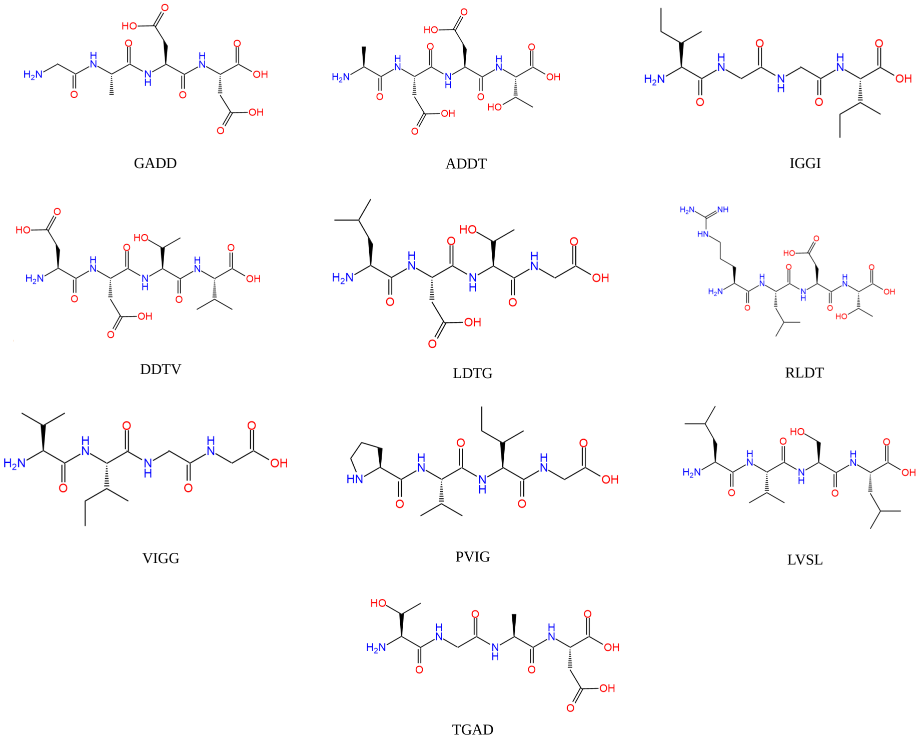


Figure 6: **Examples of the four different motif categories** that we apply in Molformer based on functional groups.

For motifs in proteins, ally, there are only 29,871 kinds of quaternions in PDBbind. Besides, we utilize a Latin hyper-cube sampling to sample 1K quaternary amino acids as the candidates in each iteration, and γ is set as $1e^{-3}$. The motif lexicon explored by our RL method for the PDBbind task is in Figure 7. Moreover, the portion of motifs in a protein (or some part such as a pocket) is the number of motifs divided by the number of all amino acids in that protein.



GADD

ADDT

IGGI

DDTV

LDTG

RLDT

VIGG

PVIG

LVSL

TGAD

Figure 7: **The motif lexicon found by our RL method,** where each motif is composed of ten quaternary amino acids and the upper-case names correspond to their compositions.

Table 5: Key statistics of datasets from three different categories.

Category	Dataset	Tasks	Task Type	Molecules	Atom Class	Min. Atoms	Max. Atoms	Density (Å)	Metric
Quantum Chemistry	QM7	1	regression	7,160	5	4	23	2.91	MAE
	QM8	12	regression	21,786	5	3	26	1.54	MAE
	QM9	12	regression	133,885	5	3	28	1.61	MAE
Physiology	BBBP	1	classification	2,039	13	2	132	2.64	ROC-AUC
	ClinTox	2	classification	1,478	27	1	136	2.83	ROC-AUC
Biophysics	PDBind^2	1	regression	11,908	23	115	1,085	5.89	RMSE
	BACE	1	classification	1,513	8	10	73	3.24	ROC-AUC

A.2 Data Summary

B Additional Experimental Results

B.1 Conformation Classification

Task and Data. To explore the influence of multiple conformations, we introduce a new task, conformation classification, to evaluate model's capacity to differentiate molecules with various low-energy conformations. We use the recent GEOM-QM9 [4] that is an extension to QM9 dataset. It contains multiple conformations for most molecules, while the original QM9 only contains one.

We randomly draw 1000 different molecules from GEOM-QM9, each with 20 different conformations. Models are required to distinguish the molecular type given different conformations. We take a half of each molecular conformations as the training set and another half as the test split. Since it is a multi-class classification problem with 1000 classes, we compute the micro-average and macro-average ROC-AUC as well as the accuracy for evaluations.

Results. Molformer achieves a perfect micro-average and macro-average ROC-AUC as well as a high accuracy (see Table 6). This indicates strong robustness of our model against different spatial conformations of molecules.

Table 6: Molformer performance on conformation classification.

Metrics	Acc.	Micro.	Macro.
Molformer	0.999	1.000	1.000

B.2 AFPS vs. FPS.

To have a vivid understanding of the atom sampling algorithm, we conducted a case study on a random molecule (see Figure 8). Points selected by FPS are randomized and exclude vital atoms like the heavy metal Nickel (Ni). With the adoption of AFPS, sampled points include Ni, Nitrogen (N) and the benzene ring besides that they keep remote distances from each other. Moreover, FPS integrates too many features of trivial atoms like Hydrogen (H) while misses out key atoms and motifs, which will significantly smooth the molecular representations and lead to poor predictions. This illustrative example shows the effectiveness of our AFPS to offset disadvantages of the conventional FPS in 3D molecular representation.

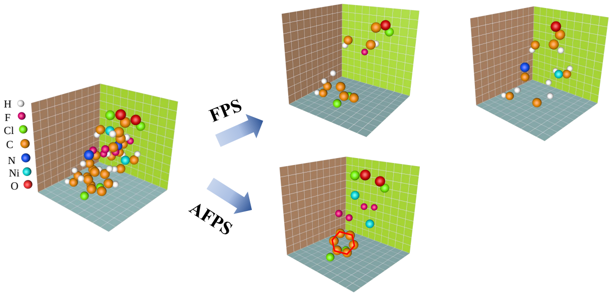


Figure 8: Sampled points using FPS and AFPS. The red circle represents a benzene ring. We do not show dummy nodes there.

B.3 Protein

We envision a protein-ligand pair in PDBbind in Figure 9. It can be observed that motifs occurs much more frequently in the area of the protein pocket than other places. To be specific, the ligand is exactly surrounded my our discovered motifs, which strongly demonstrates the effectiveness of our RL method to mine motifs with semantic meanings.

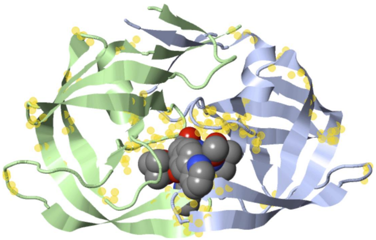


Figure 9: The protein-ligand pair of 2aqu in PDBbind. The yellow dot-halos denotes the motifs found in this protein.

[2] The total number of proteins in the full, unsplit PDBbind is 11K, but our experiment only uses 4K proteins at 30% sequence identity. Moreover, the number of atoms is the sum of both the pocket and molecules.