GeoMFormer: A General Architecture for Geometric Molecular Representation Learning

Tianlang Chen[1*] **Shengjie Luo**[2*, ✉], **Di He**[2,✉], **Shuxin Zheng**[3], **Tie-Yan Liu**[3], **Liwei Wang**[2,4,✉]

[1]School of EECS, Peking University [2]National Key Laboratory of General Artificial Intelligence, School of Intelligence Science and Technology, Peking University [3]Microsoft Research
[4]Center for Machine Learning Research, Peking University

tlchen@pku.edu.cn, luosj@stu.pku.edu.cn, dihe@pku.edu.cn
{shuz, tyliu}@microsoft.com, wanglw@pku.edu.cn

Abstract

Molecular modeling, a central topic in quantum mechanics, aims to accurately calculate the properties and simulate the behaviors of molecular systems. The molecular model is governed by physical laws, which impose geometric constraints such as invariance and equivariance to coordinate rotation and translation. While numerous deep learning approaches have been developed to learn molecular representations under these constraints, most of them are built upon heuristic and costly modules. We argue that there is a strong need for a general and flexible framework for learning both invariant and equivariant features. In this work, we introduce a novel Transformer-based molecular model called GeoMFormer to achieve this goal. Using the standard Transformer modules, two separate streams are developed to maintain and learn invariant and equivariant representations. Carefully designed *cross-attention* modules bridge the two streams, allowing information fusion and enhancing geometric modeling in each stream. As a general and flexible architecture, we show that many previous architectures can be viewed as special instantiations of GeoMFormer. Extensive experiments are conducted to demonstrate the power of GeoMFormer. All empirical results show that GeoMFormer achieves strong performance on both invariant and equivariant tasks of different types and scales. Code and models will be made publicly available.

1 Introduction

Deep learning approaches have emerged as a powerful tool for a wide range of tasks [21, 10, 5]. Recently, researchers have started investigating whether the power of neural networks could help solve problems in physics and chemistry, such as predicting the property of molecules with 3D coordinates and simulating how each atom moves in Euclidean space [51, 16, 48]. These molecular modeling tasks require the learned model to satisfy general physical laws, such as the invariance and equivariance conditions: The model's prediction should react *physically* when the input coordinates change according to the transformation of the coordinate system, such as rotation and translation.

A variety of methods have been proposed to design neural architectures that intrinsically satisfy the invariance or equivariance conditions [56, 50, 3]. To satisfy the invariant condition, several approaches incorporate invariant features, such as the relative distance between each atom pair, into classic neural networks [51, 53]. However, this may hinder the model from effectively extracting the molecular structural information. For example, computing dihedral angles from coordinates is straightforward but requires much more nonlinear operations using relative distances. To satisfy the equivariant condition, several works design neural networks with equivariant operation only, such as tensor product between irreducible representations [56, 13, 3] and vector operations [48, 50, 55]. However, the number of such operations is limited, and they are either costly to scale or lead to fairly complex network architecture designs to guarantee sufficient expressive power. More importantly, many

[*]Equal contributions.

NeurIPS 2023 AI for Science Workshop.

real-world applications require a model that can effectively perform both invariant and equivariant prediction with strong performance at the same time. While some recent works study this direction [50, 55], most proposed networks are designed heuristically and lack general design principles.

We argue that developing a general and flexible architecture to effectively learn both invariant and equivariant representations is essential. In this work, we introduce GeoMFormer to achieve this goal. GeoMFormer uses a standard Transformer-based architecture [58] but with two streams. An invariant stream learns invariant representations, and an equivariant stream learns equivariant representations. Each stream consists of invariant/equivariant self-attention and feed-forward layers. The key design in GeoMFormer is to use *cross-attention mechanisms* between the two streams, letting each stream incorporate the information from the other and enhance itself. In each layer of the invariant stream, we develop an *invariant-to-equivariant* cross-attention module, where the invariant representations are used to query key-value pairs in the equivariant stream. An *equivariant-to-invariant* cross-attention module is similarly designed for the equivariant stream. We show that the design of all self-attention and cross-attention modules is flexible and how to satisfy the invariant/equivariant conditions effectively.

Our proposed architecture has several advantages compared to previous works. GeoMFormer decomposes the invariant/equivariant representation learning through self-attention and cross-attention modules. By interacting the two streams using cross-attention modules, the invariant stream receives more structural signals (from the equivariant stream), and the equivariant stream obtains more non-linear transformation (from the invariant stream), which allows simultaneously and completely modeling interatomic interactions within/across feature spaces in a unified manner. Furthermore, we demonstrate that the proposed decomposition is general by showing that many existing methods can be regarded as special cases in our framework. For example, PaiNN[50] and TorchMD-NET[55] can be formulated as a special instantiation by following the design philosophy of GeoMFormer and using proper instantiations of key building components. From this perspective, we believe our architecture can offer many different options in different scenarios in real applications.

We evaluate our architecture on diverse datasets with both invariant and equivariant targets. On the Open Catalyst 2020 (OC20) dataset [6], which contains large atomic systems composed of an adsorbate and a catalyst, our architecture accurately predicts energy (invariant) and relaxed structure (equivariant). Additionally, t achieves state-of-the-art performance in predicting the homo-lumo energy gap (invariant) on PCQM4Mv2 [22] and Molecule3D [63] datasets, both comprising molecules from chemical databases[41, 44]. Moreover, we conduct an N-body simulation experiment, wherein our architecture precisely forecasts particle positions (equivariant) governed by physical rules. All the empirical results highlight the generality and effectiveness of our approach.

2 Related Works

Invariant Representation Learning. Recently, invariance has been recognized as one of the key principles guiding the development of molecular models. To describe the properties of a molecular system, the model's prediction should remain unchanged if we conduct any rotation or translation actions on the coordinates of the whole system. Previous works typically used relative structural information from coordinates to inherently maintain invariance. In SchNet [51], interatomic distances were encoded via radial basis functions, serving as weights for the continuous-filter convolutional layers. PhysNet [57] similarly incorporated both atomic features and interatomic distances in its interaction blocks. Graphormer-3D [53] employs a Transformer-based model, encoding relative distances as attention bias terms, which shows strong performance on large-scale datasets [6].

Beyond the interatomic distance, other works further incorporate high-order invariant signals. Based on PhysNet, DimeNet [16] and DimeNet++ [15] additionally encode the bond angle information using Fourier-Bessel basis functions. Moreover, GemNet [14] and GemNet-OC [17] carefully studied the connections between spherical representations and directional information, which inspired to leverage the dihedral angles, i.e., angles between planes formed by bonds. SphereNet [37] and ComENet [59] consider the torsional information to augment the molecular models. During the development in the literature, more complex features are incorporated due to the lossy structural information when purely learning invariant representations, while largely increasing the costs. Moreover, these invariant models are generally unable to directly perform equivariant prediction tasks.

Equivariant Representation Learning. Rather than focusing solely on invariant blocks, various works aim to learn equivariant representations. In real-world applications, numerous molecular tasks require equivariant predictions, such as predicting forces, positions, velocities, and other tensorized properties in dynamic simulations. When rotating positions, these properties should rotate correspondingly . One classical approach [56, 13, 3, 43] to encoding the equivariant constraints is using irreducible representations (irreps) via spherical harmonics [20]. Equivariant convolutions based on tensor products between irreps enable models to preserve equivariance. However, these models do not always significantly outperform invariant models on invariant tasks. Besides, their operations are in general costly [50, 48, 12], hindering deployment in large-scale molecular systems.

On the other hand, several recent works maintain both invariant and equivariant representations. EGNN [48] proposed a simple framework. Its invariant representations encode type information and relative distance, and are further used in vector scaling functions to transform the equivariant representations. PaiNN [50] extended EGNN's framework to include the Hardamard product operation to transform the equivariant representations. Based on the operations of PaiNN, TorchMD-Net [55] further proposed a modified version of the self-attention modules to update invariant representations and achieved better performance on invariant tasks. Allegro [43] used tensor product operations to update equivariant features and interacted equivariant and invariant features by using weight-generation modules. In contrast, our GeoMFormer is developed based on a general design philosophy to learn both invariant and equivariant representations, enabling simultaneous and complete modeling of interatomic interactions within/across feature spaces in a unified manner, as introduced in Section 4.1.

3 Preliminary

3.1 Notations & Geometric Constraints

We denote a molecular system as $\mathcal{M}$, which consists of a collection of atoms held together by attractive forces. Let $\mathbf{X} \in \mathbb{R}^{n\times d}$ denote the atoms with features, where n is the number of atoms, and d is the feature dimension. Given atom i, we use $\mathbf{r}_i \in \mathbb{R}^3$ to denote its cartesian coordinate in the three-dimensional Euclidean space. We define $\mathcal{M} = (\mathbf{X}, R)$, where $R = \{\mathbf{r}_1, ..., \mathbf{r}_n\}$.

In nature, molecular systems are subject to physical laws that impose geometric constraints on their properties and behaviors. For instance, if the position of each atom is translated by a constant vector in Euclidean space, the system's total energy remains unchanged. If a rotation is applied to each position, the direction of the force on each atom will rotate correspondingly. Mathematically, these geometric constraints are closely tied to the principles of invariance and equivariance in group theory. [9, 8, 52].

Formally, let $\phi : \mathcal{X} \to \mathcal{Y}$ denote a function mapping between vector spaces. Given a group G, let $\rho^{\mathcal{X}}$ and $\rho^{\mathcal{Y}}$ denote its group representations. A function $\phi : \mathcal{X} \to \mathcal{Y}$ is said to be equivariant/invariant if it satisfies the following conditions respectively:

$$\begin{aligned} &\textit{Equivariance}: \rho^{\mathcal{Y}}(g)[\phi(x)] = \phi\left(\rho^{\mathcal{X}}(g)[x]\right), \text{ for all } g \in G, x \in \mathcal{X} \\ &\textit{Invariance}: \qquad\quad \phi(x) = \phi\left(\rho^{\mathcal{X}}(g)[x]\right), \text{ for all } g \in G, x \in \mathcal{X} \end{aligned} \tag{1}$$

Intuitively, an equivariant function transforms the output predictably in response to input transformations, while an invariant function produces an unchanged output when transformations are applied to the input. For additional background on group theory, please refer to the appendix of [56, 1, 13].

Molecular systems are naturally located in the three-dimensional Euclidean space, and the group related to translations and rotations is known as $SE(3)$. For each element g in the $SE(3)$ group, its representation on $\mathbb{R}^3$ can be parameterized by pairs of translation vectors $\mathbf{t} \in \mathbb{R}^3$ and orthogonal transformation matrices $\mathbf{R} \in \mathbb{R}^{3\times 3}, \det(\mathbf{R}) = 1$, i.e., $g = (\mathbf{t}, \mathbf{R})$. Given a vector $x \in \mathbb{R}^3$, we have $\rho^{\mathbb{R}^3}(g)[x] := \mathbf{R}x + \mathbf{t}$. For molecular modeling, it is essential to learn molecular representations that encode the rotation equivariance and translation invariance constraints. Formally, let $V_{\mathcal{M}}$ denote the space of molecular systems, for each atom i, we define equivariant representation ϕ^E and invariant representation ϕ^I if $\forall\, g = (\mathbf{t}, \mathbf{R}) \in SE(3), \mathcal{M} = (\mathbf{X}, R) \in V_{\mathcal{M}}$, the following conditions are satisfied:

$$\begin{aligned} &\phi^E : V_{\mathcal{M}} \to \mathbb{R}^{3\times d}, && \mathbf{R}\phi^E(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^E(\mathbf{X}, \{\mathbf{R}\mathbf{r}_1, ..., \mathbf{R}\mathbf{r}_n\}) \\ &\phi^E : V_{\mathcal{M}} \to \mathbb{R}^{3\times d}, && \phi^E(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^E(\mathbf{X}, \{\mathbf{r}_1 + \mathbf{t}, ..., \mathbf{r}_n + \mathbf{t}\}) \\ &\phi^I : V_{\mathcal{M}} \to \mathbb{R}^{d}, && \phi^I(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^I(\mathbf{X}, \{\mathbf{R}\mathbf{r}_1 + \mathbf{t}, ..., \mathbf{R}\mathbf{r}_n + \mathbf{t}\}) \end{aligned} \tag{2}$$

3.2 Attention module

The attention module lies at the core of the Transformer architecture [58], and it is formulated as querying a dictionary with key-value pairs, e.g., $\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d}})V$, where d is the hidden dimension, and Q (Query), K (Key), V (Value) are specified as the hidden representations of the previous layer. The multi-head variant of the attention module is widely used, as it allows the model to jointly attend to information from different representation subspaces. It is defined as follows:

$$
\begin{aligned}
\text{Multi-head}(Q, K, V) &= \text{Concat}(\text{head}_1, \cdots, \text{head}_H)W^O \\
\text{head}_k &= \text{Attention}(QW_k^Q, KW_k^K, VW_k^V),
\end{aligned}
\tag{3}
$$

where $W_k^Q \in \mathbb{R}^{d\times d_H}, W_k^K \in \mathbb{R}^{d\times d_H}, W_k^V \in \mathbb{R}^{d\times d_H}$, and $W^O \in \mathbb{R}^{Hd_H\times d}$ are learnable matrices, H is the number of heads. d_H is the dimension of each attention head.

Serving as a generic building block, the attention module can be used in various ways. On the one hand, the self-attention module specifies Query, Key, and Value as the same hidden representation, thereby extracting contextual information for the input. It has been one of the key components in Transformer-based foundation models across various domains [10, 5, 11, 38, 64, 28]. On the other hand, the cross-attention module specifies the hidden representation from one space as Query, and the representation from the other space as Key-Value pairs, e.g. encoder-decoder attention for sequence-to-sequence learning. As the cross-attention module bridges two representation spaces, it has been also widely used beyond Transformer for information fusion and improving representations [33, 24, 27, 26].

4 GeoMFormer

In this section, we introduce GeoMFormer, a novel Transformer-based molecular model for learning invariant and equivariant molecular representations. We begin by elaborating on the key designs of GeoMFormer, which form a general framework to guide the development of geometric molecular models (Section 4.1), Next we thoroughly discuss the implementation details of GeoMFormer (Section 4.2).

4.1 A General Design Philosophy

As previously mentioned, several existing works learned invariant representations using invariant features, such as distance information, which may have difficulty in extracting other useful structural signals. Some other works developed equivariant models via equivariant operations, which are either heuristic or costly. Instead, we aim to develop a general design principle, which guides the development of a model instance that addresses the disadvantages aforementioned in both invariant and equivariant representation learning.

We call our model GeoMFormer, which is a two-stream Transformer model to encode invariant and equivariant information. Each stream is built up using stacked Transformer blocks, each of which consists of a self-attention module and a cross-attention module, followed by a feed-forward network. For each atom $k \in [n]$, we use $\mathbf{z}_k^I \in \mathbb{R}^d$ and $\mathbf{z}_k^E \in \mathbb{R}^{3\times d}$ to denote its invariant and equivariant representations respectively. Let $\mathbf{Z}^I = [\mathbf{z}_1^{I^\top}; ...; \mathbf{z}_n^{I^\top}] \in \mathbb{R}^{n\times d}$ and $\mathbf{Z}^E = [\mathbf{z}_1^E; ...; \mathbf{z}_n^E] \in \mathbb{R}^{n\times 3\times d}$, the invariant (colored in red) and equivariant (colored in blue) representations are updated in the following manner:

$$
\begin{aligned}
&\textit{Invariant Stream}\begin{cases}
\mathbf{Z}'^{I,l} &= \mathbf{Z}^{I,l} + \text{Inv-Self-Attn}(\mathbf{Q}^{I,l}, \mathbf{K}^{I,l}, \mathbf{V}^{I,l}) \\
\mathbf{Z}''^{I,l} &= \mathbf{Z}'^{I,l} + \text{Inv-Cross-Attn}(\mathbf{Q}^{I,l}, \mathbf{K}^{I\_E,l}, \mathbf{V}^{I\_E,l}) \\
\mathbf{Z}^{I,l+1} &= \mathbf{Z}''^{I,l} + \text{Inv-FFN}(\mathbf{Z}''^{I,l})
\end{cases} \\
&\textit{Equivariant Stream}\begin{cases}
\mathbf{Z}'^{E,l} &= \mathbf{Z}^{E,l} + \text{Equ-Self-Attn}(\mathbf{Q}^{E,l}, \mathbf{K}^{E,l}, \mathbf{V}^{E,l}) \\
\mathbf{Z}''^{E,l} &= \mathbf{Z}'^{E,l} + \text{Equ-Cross-Attn}(\mathbf{Q}^{E,l}, \mathbf{K}^{E\_I,l}, \mathbf{V}^{E\_I,l}) \\
\mathbf{Z}^{E,l+1} &= \mathbf{Z}''^{E,l} + \text{Equ-FFN}(\mathbf{Z}''^{E,l})
\end{cases}
\end{aligned}
\tag{4}
$$

where l denotes the layer index. In this framework, the self-attention modules and feed-forward networks are used to iteratively update representations in each stream. The cross-attention modules use representations from one stream to query Key-Value pairs from the other stream. By using this mechanism, a bidirectional bridge is established between invariant and equivariant streams. Besides the contextual information from the invariant stream itself, the invariant representations can freely attend to more geometrical signals from the equivariant stream. Similarly, the equivariant representations can benefit from using more non-linear transformations in the invariant representations. With the cross-attention modules, the expressiveness of both invariant and equivariant representation learning is largely improved, which allows simultaneously and completely modeling interatomic interactions within/across feature spaces in a unified manner. In this regard, as highlighted by different colors, the Query, Key, and Value in the self-attention modules (Inv-Self-Attn,Equ-Self-Attn) and the cross-attention modules (Inv-Cross-Attn,Equ-Cross-Attn) are differently specified, which should carefully encode the geometric constraints mentioned in Section 3.1, as introduced below.

Desiderata for Invariant Self-Attention. Given the invariant representation $\mathbf{Z}^I$, the Query, Key and Value in $\mathrm{Inv\text{-}Self\text{-}Attn}$ are calculated via a function mapping $\psi^I : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d}$, i.e., $\mathbf{Q}^I = \psi^I_Q(\mathbf{Z}^I), \mathbf{K}^I = \psi^I_K(\mathbf{Z}^I), \mathbf{V}^I = \psi^I_V(\mathbf{Z}^I)$. Essentially, the attention module linearly transforms the Value $\mathbf{V}^I$, with the weights being calculated from the dot product between the Query and Key (i.e., attention scores). In this regard, if both $\mathbf{V}^I$ and the attention scores preserve the invariance, then the output satisfies the invariant constraint, i.e., ψ^I is required to be invariant. Under this condition, it is easy to check the output representation of this module keeps the invariance, which is proved in the appendix.

Desiderata for Equivariant Self-Attention. Similarly, given the equivariant input $\mathbf{Z}^E$, the Query, Key and Value in $\mathrm{Equ\text{-}Self\text{-}Attn}$ are calculated via a function mapping $\psi^E : \mathbb{R}^{n\times 3\times d} \to \mathbb{R}^{n\times 3\times d}$, i.e., $\mathbf{Q}^E = \psi^E_Q(\mathbf{Z}^E), \mathbf{K}^E = \psi^E_K(\mathbf{Z}^E), \mathbf{V}^E = \psi^E_V(\mathbf{Z}^E)$. Similarly, ψ^E is required to be equivariant. However, this still cannot guarantee the module to be equivariant if standard attention is used. We modified $\alpha_{ij} = \sum_{k=1}^{d} \mathbf{Q}^E{}_{[i,:,k]} \mathbf{K}^E{}_{[j,:,k]}{}^\top$, where $\mathbf{Q}^E{}_{[i,:,k]} \in \mathbb{R}^3$ denotes the k-th dimension of the atom i's Query. It is straightforward to check the equivariance is preserved, which is proved in the appendix.

Desiderata for Cross-attentions between the two Streams. In each stream, the cross-attention module is used to leverage information from the other stream. We call the cross attention in the invariant stream *invariant-cross-equivariant* attention, and call the cross attention in the equivariant stream *equivariant-cross-invariant* attention, i.e., $\mathrm{Inv\text{-}Cross\text{-}Attn}$ and $\mathrm{Equ\text{-}Cross\text{-}Attn}$. The difference between the two cross attention lies in how the query, key, value are specified:

$$\begin{aligned} &\textit{Invariant-cross-Equivariant} \quad \mathbf{Q}^{I\_E} = \psi^I_Q(\mathbf{Z}^I), \mathbf{K}^{I\_E} = \psi^{I\_E}_K(\mathbf{Z}^I, \mathbf{Z}^E), \mathbf{V}^{I\_E} = \psi^{I\_E}_V(\mathbf{Z}^I, \mathbf{Z}^E) \\ &\textit{Equivariant-cross-Invariant} \quad \mathbf{Q}^{E\_I} = \psi^E_Q(\mathbf{Z}^E), \mathbf{K}^{E\_I} = \psi^{E\_I}_K(\mathbf{Z}^E, \mathbf{Z}^I), \mathbf{V}^{E\_I} = \psi^{E\_I}_V(\mathbf{Z}^E, \mathbf{Z}^I) \end{aligned} \tag{5}$$

First, for Query $\mathbf{Q}^{I\_E}$ and $\mathbf{Q}^{E\_I}$, the requirement to ψ^I and ψ^E remains the same as previously stated. Moreover, as distinguished by different colors, the Key-Value pairs and the Query are calculated in different ways, for which the requirement should be separately considered. Note that both $\mathbf{V}^{I\_E}$ and $\mathbf{V}^{E\_I}$ are still linearly transformed by the cross-attention modules. If $\mathbf{V}^{I\_E}$ preserves the invariance and $\mathbf{V}^{E\_I}$ preserves the equivariance, then the remaining condition is to keep the invariance of the attention score calculation. That is to say, for the $\mathrm{Inv\text{-}Cross\text{-}Attn}$, both ψ^I and ψ^{I_E} are required to be invariant. It is similar to the $\mathrm{Equ\text{-}Cross\text{-}Attn}$ that both ψ^E and ψ^{E_I} are required to be equivariant. In this way, the outputs of both cross-attention modules are under the corresponding geometric constraints, which is proved in the appendix.

Discussion. The carefully designed blocks outlined above provide a general design philosophy for encoding the geometric constraints and bridging the invariant and equivariant molecular representations, which lie at the core of our framework. Note that the translation invariance can be easily preserved by encoding relative structure signals of the input. It is also worth pointing out that we do not restrict the specific instantiation of each component, and various design choices can be adopted as long as they meet the requirements mentioned above. Moreover, we prove that our framework can include many previous models as an instantiation, e.g., PaiNN [50] and TorchMD-Net [55], can be extended to encode additional geometric constraints [8], which are presented in the appendix. In this work, we present a simple yet effective model instance that implements this design philosophy, which we will thoroughly introduce in the next subsection.

4.2 Implementation Details of GeoMFormer

Following the design guidance in Section 4.1, we propose **Geo**metric **M**olecular Transformer (GeoMFormer). The overall architecture of GeoMFormer is shown in Figure 1, which is composed of stacked GeoMFormer blocks (Eqn.(5)). We introduce the instantiations of the self-attention, cross-attention and FFN modules below and prove the properties they satisfy in the appendix. We also incorporate widely used modules like Layer Normalization [2] and Structural Encodings [53] for better empirical performance. Due to the space limits, we refer readers to the appendix for further details.

Instantiation of Self-Attention. In GeoMFormer, the linear function is used to implement both $\psi^I : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d}$ and $\psi^E : \mathbb{R}^{n\times 3\times d} \to \mathbb{R}^{n\times 3\times d}$:

$$\begin{aligned} &\mathbf{Q}^I = \psi^I_Q(\mathbf{Z}^I) = \mathbf{Z}^I W^I_Q, \quad \mathbf{K}^I = \psi^I_K(\mathbf{Z}^I) = \mathbf{Z}^I W^I_K, \quad \mathbf{V}^I = \psi^I_V(\mathbf{Z}^I) = \mathbf{Z}^I W^I_V \\ &\mathbf{Q}^E = \psi^E_Q(\mathbf{Z}^E) = \mathbf{Z}^E W^E_Q, \quad \mathbf{K}^E = \psi^E_K(\mathbf{Z}^E) = \mathbf{Z}^E W^E_K, \quad \mathbf{V}^E = \psi^E_V(\mathbf{Z}^E) = \mathbf{Z}^E W^E_V \end{aligned} \tag{6}$$

where $W^I_Q, W^I_K, W^I_V, W^E_Q, W^E_K, W^E_V \in \mathbb{R}^{d\times d_H}$ are learnable parameters.

Instantiation of Cross-Attention. As previously stated, both ψ^{I_E} and ψ^{E_I} in the cross-attention modules fuse representations from different spaces (invariant & equivariant) into target spaces. In the

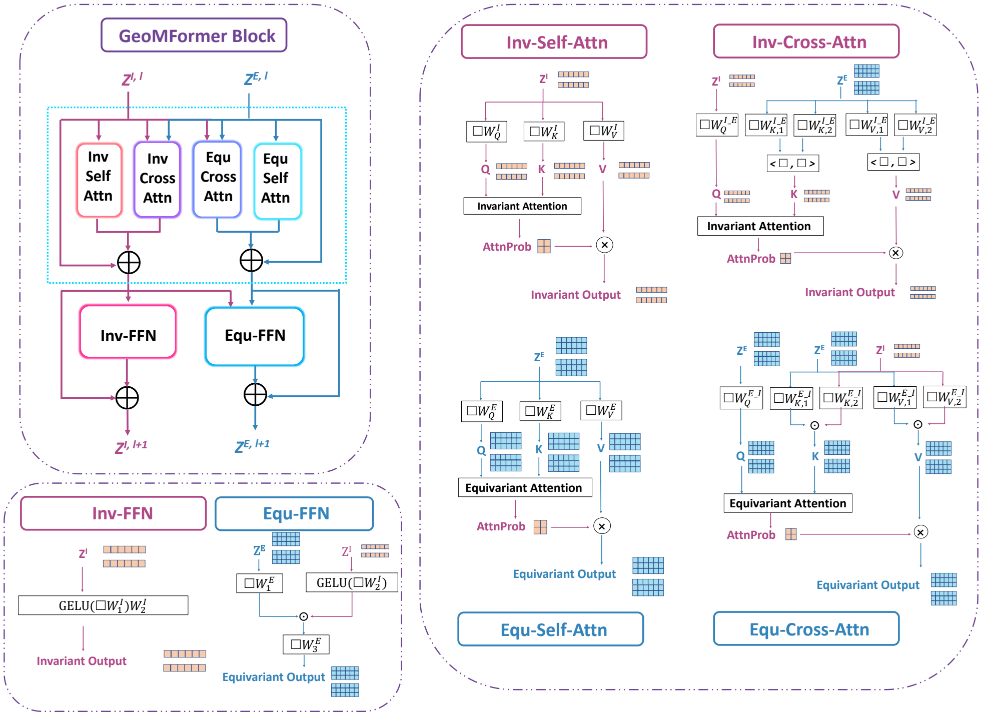

Figure 1: An illustration of our GeoMFormer model architecture.

Invariant-cross-Equivariant attention module (Inv-Cross-Attn), to obtain the Key-Value pairs, the equivariant representations are mapped to the invariant space. For the sake of simplicity, we use the dot-product operation $< \cdot, \cdot >$ to instantiate ψ^{I_E}. Given $X, Y \in \mathbb{R}^{n\times 3\times d}$, $Z =< X, Y >\in \mathbb{R}^{n\times d}$, where $Z_{[i,k]} = X_{[i,:,k]}{}^\top Y_{[i,:,k]}$. Then the Key-Value pairs in Inv-Cross-Attn are calculated as:

$$\mathbf{K}^{I\_E} = \psi_K^{I\_E}(\mathbf{Z}^I, \mathbf{Z}^E) =< \mathbf{Z}^E W_{K,1}^{I\_E}, \mathbf{Z}^E W_{K,2}^{I\_E} >, \quad \mathbf{V}^{I\_E} = \psi_V^{I\_E}(\mathbf{Z}^I, \mathbf{Z}^E) =< \mathbf{Z}^E W_{V,1}^{I\_E}, \mathbf{Z}^E W_{V,2}^{I\_E} > \tag{7}$$

where $W_{K,1}^{I\_E}, W_{K,2}^{I\_E}, W_{V,1}^{I\_E}, W_{V,2}^{I\_E} \in \mathbb{R}^{d\times d_H}$ for Key and Value are learnable parameters. On the other hand, the invariant representations are mapped to the equivariant space in the *Equivariant-cross-Invariant* attention module (Equ-Cross-Attn). To achieve this goal, we use the scalar product $\odot$ to instantiate ψ^{E_I}. Given $X \in \mathbb{R}^{n\times 3\times d}, Y \in \mathbb{R}^{n\times d}$, $Z = X \odot Y \in \mathbb{R}^{n\times 3\times d}$, where $Z_{[i,j,k]} = X_{[i,j,k]} \cdot Y_{[i,k]}$. Using this operation, the Key-Value pairs in Equ-Cross-Attn are calculated as:

$$\mathbf{K}^{E\_I} = \psi_K^{E\_I}(\mathbf{Z}^E, \mathbf{Z}^I) = \mathbf{Z}^E W_{K,1}^{E\_I} \odot \mathbf{Z}^I W_{K,2}^{E\_I}, \quad \mathbf{V}^{E\_I} = \psi_V^{E\_I}(\mathbf{Z}^E, \mathbf{Z}^I) = \mathbf{Z}^E W_{V,1}^{E\_I} \odot \mathbf{Z}^I W_{V,2}^{E\_I} \tag{8}$$

where $W_{K,1}^{E\_I}, W_{K,2}^{E\_I}, W_{V,1}^{E\_I}, W_{V,2}^{E\_I} \in \mathbb{R}^{d\times d_H}$ are learnable parameters.

Instantiation of Feed-Forward Networks. Feed-forward networks (FFN) also play important roles in refining contextual representations. In the invariant stream, the FFN is kept unchanged from the standard Transformer model, i.e., Inv-FFN$(\mathbf{Z}''^I) = \mathrm{GELU}(\mathbf{Z}''^I W_1^I)W_2^I$, where $W_1^I \in \mathbb{R}^{d\times r}, W_2^I \in \mathbb{R}^{r\times d}$ and r denotes the hidden dimension of the FFN layer. In the equivariant stream, it is worth noting that commonly used non-linear activation functions break the equivariant constraints. In our GeoMFormer, we use the invariant representations as a gating function to non-linearly activate the equivariant representations, i.e., Equ-FFN$(\mathbf{Z}''^E) = (\mathbf{Z}''^E W_1^E \odot \mathrm{GELU}(\mathbf{Z}''^I W_2^I))W_3^E$, where $W_1^E, W_1^I \in \mathbb{R}^{d\times r}, W_2^E \in \mathbb{R}^{r\times d}$.

Input Layer. Given a molecular system $\mathcal{M} = (\mathbf{X}, R)$, we set the invariant representation at the input as $\mathbf{Z}^{I,0} = \mathbf{X}$, where $\mathbf{X}_i \in \mathbb{R}^d$ is a learnable embedding vector indexed by the atom i's type. For the equivariant representation, we set $\mathbf{Z}_i^{E,0} = \hat{\mathbf{r}}_i' g(||\mathbf{r}_i'||)^\top \in \mathbb{R}^{3\times d}$, where we consider both the direction $\hat{\mathbf{r}}_i' \in \mathbb{R}^3$ and the scale $g(||\mathbf{r}_i'||) \in \mathbb{R}^d$ of the each atom's mean-centered position $\mathbf{r}_i'$. $g : \mathbb{R} \to \mathbb{R}^d$ is instantiated by the Gaussian Basis Kernel, i.e., $g(||\mathbf{r}_i'||) = \psi_i W$, $\psi_i = [\psi_i^1; ...; \psi_i^d]^\top$, $\psi_i^k = -\frac{1}{\sqrt{2\pi}|\sigma^k|}\exp\left(-\frac{1}{2}\left(\frac{\gamma_i||\mathbf{r}_i'||+\beta_i-\mu^k}{|\sigma^k|}\right)^2\right), k = 1, ..., d$, where $W \in \mathbb{R}^{d\times d}$ is learnable, γ_i, β_i

are learnable scalars indexed by the atom type, and μ^k, σ^k are learnable kernel center and scaling factor of the k-th Kernel. Note that our GeoMFormer is not restricted to these choices, which can encode additional features if the constraints are satisfied, as discussed in the appendix.

5 Experiments

We empirically evaluate our GeoMFormer on extensive tasks, covering different types (invariant & equivariant), data (simple molecules & adsorbate-catalyst complexes & particle systems), and scales, as shown in Table 1. Due to space limits, we present more results and ablation studies in Appendix D.

Table 1: Summarization of empirical evaluation setup.

Dataset	**Task Description**	**Task Type**	**Data Type**	**Training set size**
OC20, IS2RE [6]	Equilibrium Energy Prediction (Sec 5.1.1)	Invariant	Adsorbate-Catalyst complex	460,328
OC20, IS2RS [6]	Equilibrium Structure Prediction (Sec 5.1.2)	Equivariant	Adsorbate-Catalyst complex	460,328
PCQM4Mv2 [22]	HOMO-LUMO Gap Prediction (Sec 5.2)	Invariant	Simple molecule	3,378,606
N-Body Simulation [48]	Position Prediction (Sec 5.3)	Equivariant	Particle System	3,000
Molecule3D [59]	HOMO-LUMO Gap Prediction (Sec D.1)	Invariant	Simple molecule	2,339,788
MD17 [7]	Force Field Modeling (Sec D.2)	Inv/Equ	Simple molecule	950

5.1 OC20 Performance (Invariant & Equivariant)

The Open Catalyst 2020 (OC20) dataset [6] was created for catalyst discovery. Each data is in the form of adsorbate-catalyst complex. We focus on the tasks Initial Structure to Relaxed Energy (IS2RE) and Initial Structure to Relaxed Structure (IS2RS), which require a model to directly predict the relaxed energy and structure given the initial structure as input respectively. we use direct prediction setting instead of iterative relaxation setting, which is efficient yet more challenging. The validation sets consider the in-distribution (ID) and out-of-distribution settings that uses unseen adsorbates (OOD-Ads), catalysts (OOD-Cat) or both (OOD-Both).

5.1.1 IS2RE Performance (Invariant)

This task evaluates the model's proficiency in learning invariant representations. We follow the experimental setup of Graphormer-3D [53]. The metric is the Mean Absolute Error (MAE) and the percentage of data instances where the predicted energy is within a 0.02 eV threshold (EwT). Due to space limits, the detailed description of training settings and baselines is presented in the appendix. The results are shown in Table 2. Our GeoMFormer outperforms the compared baselines significantly, achieving 6.1% relative MAE reduction and 42.2% relative EwT improvement than previous best model. The results indeed demonstrate the effectiveness of our GeoMFormer on learning invariant representations.

Table 2: Results on IS2RE validation set. We report official results of baselines from original paper.

	Energy MAE (eV) ↓					EwT (%) ↑				
Model	ID	OOD Ads.	OOD Cat.	OOD Both	Average	ID	OOD Ads.	OOD Cat.	OOD Both	Average
CGCNN [60]	0.6203	0.7426	0.6001	0.6708	0.6585	3.36	2.11	3.53	2.29	2.82
SchNet [51]	0.6465	0.7074	0.6475	0.6626	0.6660	2.96	2.22	3.03	2.38	2.65
DimeNet++ [15]	0.5636	0.7127	0.5612	0.6492	0.6217	4.25	2.48	4.4	2.56	3.42
GemNet-T [14]	0.5561	0.7342	0.5659	0.6964	0.6382	4.51	2.24	4.37	2.38	3.38
SphereNet [37]	0.5632	0.6682	0.5590	0.6190	0.6024	4.56	2.70	4.59	2.70	3.64
Graphormer-3D [53]	0.4329	0.5850	0.4441	0.5299	0.4980	-	-	-	-	-
GNS [46]	0.47	0.51	0.48	0.46	0.4800	-	-	-	-	-
Equiformer [35]	0.4156	0.4976	0.4165	0.4344	0.4410	7.47	4.64	7.19	4.84	6.04
GeoMFormer (ours)	**0.3883**	**0.4562**	**0.4037**	**0.4083**	**0.4141**	**11.26**	**6.70**	**9.97**	**6.42**	**8.59**

5.1.2 IS2RS Performance (Equivariant)

We evaluate the model's ability to perform equivariant prediction by IS2RS task. The metric is Average Distance within Threshold (ADwT) across different thresholds. The Distance within Threshold is computed as the percentage of structures with atom position MAE below the threshold. We re-implement several competitive baselines under direct prediction setting. Please refer to the appendix for detailed settings. From Table 3, we can see that the IS2RS task under direct prediction setting is very difficult. The compared baselines consistently achieve low ADwT. Our GeoMFormer achieves the best, which indeed verifies its superior ability to perform equivariant molecular tasks.

Table 3: Results on IS2RS validation set. All models follow the direct prediction setting.

	ADwT (%) ↑				
Model	ID	OOD Ads	OOD Cat	OOD Both	Average
PaiNN [50]	3.29	2.37	3.10	2.33	2.77
TorchMD-Net [55]	3.32	3.35	2.94	2.89	3.13
Spinconv [54]	5.81	4.88	5.63	4.84	5.29
GemNet-dT [14]	6.87	7.10	6.03	7.08	6.77
GemNet-OC [17]	11.31	**12.20**	4.40	5.55	8.36
GeoMFormer (ours)	**11.45**	10.52	**9.94**	**10.78**	**10.67**

5.2 PCQM4Mv2 Performance (Invariant)

PCQM4Mv2 is one of the largest quantum chemical property datasets from the OGB Large-Scale Challenge ([22]). The task involves predicting the HOMO-LUMO energy gap of a molecule's equilibrium structure, evaluating the model's capacity for invariant prediction. This property holds significance in real applications such as reactivity. Ground-truth labels are derived from DFT calculations. The total number of training samples is around 3.37 million.

In a practical setting, the DFT-calculated equilibrium geometric structure of each training sample is provided, but only initial structure is available for each validation sample. In this regard, we adopt one recent approach (Uni-Mol+ [39]) to handle this task. During training, the model receives efficient but inaccurate RDKit-generated[32] structures as input, and predicts both the HOMO-LUMO energy gap and the equilibrium structure by using both invariant and equivariant representations. After training, the model can be used to predict the HOMO-LUMO gap target by only using the initial structure, which meets the requirement of the settings. We compare to various baselines in the leaderboard. More details are presented in the appendix.

Table 4: Results on PCQM4Mv2. The evaluation metric is the Mean Absolute Error (MAE). We report the official results of baselines. $*$ indicates the best performance achieved by models with the same complexity (n denotes the number of atoms).

Model	Complexity	# param.	Valid MAE ↓
MLP-Fingerprint [22]	$\mathcal{O}(n)$	16.1M	0.1735
GINE-$_{\text{VN}}$ [4, 18]		13.2M	0.1167
GCN-$_{\text{VN}}$ [30, 18]		4.9M	0.1153
GIN-$_{\text{VN}}$ [62, 18]		6.7M	0.1083
DeeperGCN-$_{\text{VN}}$ [34, 18]		25.5M	0.1021*
TokenGT [29]	$\mathcal{O}(n^2)$	48.5M	0.0910
EGT [25]		89.3M	0.0869
GRPE [45]		46.2M	0.0867
Graphormer [64, 53]		47.1M	0.0864
GraphGPS [47]		19.4M	0.0858
GPS++ [42]		44.3M	0.0778
Transformer-M [40]		47.1M	0.0787
GEM-2 [36]	$\mathcal{O}(n^3)$	32.1M	0.0793
Uni-Mol+ [39]		52.4M	0.0708*
GeoMFormer (ours)	$\mathcal{O}(n^2)$	54.5M	0.0734*

From Table 4, our GeoMFormer achieves the lowest MAE among the quadratic models, specifically, 6.7$\%$ relative MAE reduction compared to the previous best model. Besides, compared to the best model Uni-Mol+ [39], our GeoMFormer achieves competitive performance while keeping the efficiency ($\mathcal{O}(n^2)$ complexity), which can be more applicable to large molecular systems. Overall, the results further verify the effectiveness of GeoMFormer on invariant representation learning.

5.3 N-Body Simulation Performance (Equivariant)

Simulating dynamical systems consisting of a set of geometric objects interacting under physical laws is crucial in many applications, e.g. molecular dynamic simulation. Following [13, 48], we use a synthetic n-body system simulation task as an extension of molecular modeling tasks. It requires the model to predict positions of a set of particles and evaluates the model's ability for equivariant predictions. The simulated system consists of 5 particles, and each carries a positive or negative charge and has an initial position and velocity. The system adheres to physical rules involving attractive and repulsive forces. The dataset contains 3000 trajectories for training, and 2000 trajectories for validation and testing respectively. We compare several competitive baselines following [48]. Detailed descriptions of data generation, training settings, and baselines can be found in the appendix. The results are shown in Table 5. Our GeoMFormer achieves the best performance compared to all baselines. In particular, the significant 33.8% MSE reduction indeed demonstrates the GeoMFormer's superior ability on learning equivariant representations.

Table 5: Results on N-body Simulation experiment. We report the official results of baselines.

Model	MSE ↓
SE(3) Transformer [13]	0.0244
Tensor Field Network [56]	0.0155
Graph Neural Network [18]	0.0107
Radial Field [31]	0.0104
EGNN [48]	0.0071
GeoMFormer (ours)	**0.0047**

6 Conclusion

In this paper, we propose a general and flexible architecture, called GeoMFormer, for learning geometric molecular representations. Using the standard Transformer backbone, two streams are developed for learning invariant and equivariant representations respectively. In particular, the cross-attention mechanism is used to bridge these two streams, letting each stream leverage contextual information from the other stream and enhance its representations. This simple yet effective design significantly boosts both invariant and equivariant modeling. Within the newly proposed framework, many existing methods can be regarded as special instances, showing the generality of our method. We conduct extensive experiments covering diverse tasks, data and scales. All the empirical results show that our GeoMFormer can achieve strong performance in different scenarios. The potential of our GeoMFormer can be further explored in a broad range of applications in molecular modeling.

Acknowledgements

This work is supported by National Key R&D Program of China (2022ZD0114900) and National Science Foundation of China (NSFC62276005). This work is partially supported by the Shanghai Committee of Science and Technology (Grant No. 21DZ1100100). The authors also thank all reviewers for their valuable suggestions.

References

[1] Brandon Anderson, Truong Son Hy, and Risi Kondor. Cormorant: Covariant molecular neural networks. *Advances in neural information processing systems*, 32, 2019.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[3] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):2453, 2022.

[4] Rémy Brossard, Oriel Frigo, and David Dehaene. Graph convolutions that can finally model local structure. *arXiv preprint arXiv:2011.15069*, 2020.

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[6] Lowik Chanussot, Abhishek Das, Siddharth Goyal, Thibaut Lavril, Muhammed Shuaibi, Morgane Riviere, Kevin Tran, Javier Heras-Domingo, Caleb Ho, Weihua Hu, et al. Open catalyst 2020 (oc20) dataset and community challenges. *Acs Catalysis*, 11(10):6059–6072, 2021.

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

[8] John F Cornwell. *Group theory in physics: An introduction*. Academic press, 1997.

[9] F Albert Cotton. *Chemical applications of group theory*. John Wiley & Sons, 1991.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[12] Thorben Frank, Oliver Unke, and Klaus-Robert Müller. So3krates: Equivariant attention for interactions on arbitrary length-scales in molecular systems. *Advances in Neural Information Processing Systems*, 35:29400–29413, 2022.

[13] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural Information Processing Systems*, 33:1970–1981, 2020.

[14] Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34:6790–6802, 2021.

[15] Johannes Gasteiger, Shankari Giri, Johannes T Margraf, and Stephan Günnemann. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. *arXiv preprint arXiv:2011.14115*, 2020.

[16] Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

[17] Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ulissi, C Lawrence Zitnick, and Abhishek Das. Gemnet-oc: developing graph neural networks for large and diverse molecular simulation datasets. *arXiv preprint arXiv:2204.02782*, 2022.

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

[19] Jonathan Godwin, Michael Schaarschmidt, Alexander Gaunt, Alvaro Sanchez-Gonzalez, Yulia Rubanova, Petar Veličković, James Kirkpatrick, and Peter Battaglia. Simple gnn regularisation for 3d molecular property prediction & beyond. *arXiv preprint arXiv:2106.07971*, 2021.

[20] Roe Goodman and Nolan R Wallach. *Representations and invariants of the classical groups*. Cambridge University Press, 2000.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. Ogb-lsc: A large-scale challenge for machine learning on graphs. *arXiv preprint arXiv:2103.09430*, 2021.

[23] 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. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[24] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 603–612, 2019.

[25] Md Shamim Hussain, Mohammed J Zaki, and Dharmashankar Subramanian. Global self-attention as a replacement for graph convolution. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 655–665, 2022.

[26] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*, 2021.

[27] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021.

[28] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[29] Jinwoo Kim, Tien Dat Nguyen, Seonwoo Min, Sungjun Cho, Moontae Lee, Honglak Lee, and Seunghoon Hong. Pure transformers are powerful graph learners. *arXiv preprint arXiv:2207.02505*, 2022.

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

[31] Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: sampling configurations for multi-body systems with symmetric energies. *arXiv preprint arXiv:1910.00753*, 2019.

[32] Greg Landrum. Rdkit: Open-source cheminformatics software. *Github*, 2016.

[33] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European conference on computer vision (ECCV)*, pages 201–216, 2018.

[34] Guohao Li, Chenxin Xiong, Ali Thabet, and Bernard Ghanem. Deepergcn: All you need to train deeper gcns. *arXiv preprint arXiv:2006.07739*, 2020.

[35] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *The Eleventh International Conference on Learning Representations*, 2022.

[36] Lihang Liu, Donglong He, Xiaomin Fang, Shanzhuo Zhang, Fan Wang, Jingzhou He, and Hua Wu. Gem-2: Next generation molecular property prediction network by modeling full-range many-body interactions, 2022.

[37] Yi Liu, Limei Wang, Meng Liu, Yuchao Lin, Xuan Zhang, Bora Oztekin, and Shuiwang Ji. Spherical message passing for 3d molecular graphs. In *International Conference on Learning Representations (ICLR)*, 2022.

[38] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

[39] Shuqi Lu, Zhifeng Gao, Di He, Linfeng Zhang, and Guolin Ke. Highly accurate quantum chemical property prediction with uni-mol+, 2023.

[40] Shengjie Luo, Tianlang Chen, Yixian Xu, Shuxin Zheng, Tie-Yan Liu, Liwei Wang, and Di He. One transformer can understand both 2d & 3d molecular data. *arXiv preprint arXiv:2210.01765*, 2022.

[41] Nakata Maho. The pubchemqc project: A large chemical database from the first principle calculations. In *AIP conference proceedings*, volume 1702, page 090058. AIP Publishing LLC, 2015.

[42] Dominic Masters, Josef Dean, Kerstin Klaser, Zhiyi Li, Samuel Maddrell-Mander, Adam Sanders, Hatem Helal, Deniz Beker, Ladislav Rampavsek, and D. Beaini. Gps++: An optimised hybrid mpnn/transformer for molecular property prediction. *ArXiv*, abs/2212.02229, 2022.

[43] Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics. *Nature Communications*, 14(1):579, 2023.

[44] Maho Nakata and Tomomi Shimazaki. Pubchemqc project: a large-scale first-principles electronic structure database for data-driven chemistry. *Journal of chemical information and modeling*, 57(6):1300–1308, 2017.

[45] Wonpyo Park, Woong-Gi Chang, Donggeon Lee, Juntae Kim, et al. Grpe: Relative positional encoding for graph transformer. In *ICLR2022 Machine Learning for Drug Discovery*, 2022.

[46] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W Battaglia. Learning mesh-based simulation with graph networks. *arXiv preprint arXiv:2010.03409*, 2020.

[47] Ladislav Rampavsek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. *arXiv preprint arXiv:2205.12454*, 2022.

[48] Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International Conference on Machine Learning*, page 9323–9332. PMLR, 2021.

[49] Bernhard Scholkopf, Kah-Kay Sung, Christopher JC Burges, Federico Girosi, Partha Niyogi, Tomaso Poggio, and Vladimir Vapnik. Comparing support vector machines with gaussian kernels to radial basis function classifiers. *IEEE transactions on Signal Processing*, 45(11):2758–2765, 1997.

[50] Kristof Schütt, Oliver Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*, pages 9377–9388. PMLR, 2021.

[51] Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, 2018.

[52] William Raymond Scott. *Group theory*. Courier Corporation, 2012.

[53] Yu Shi, Shuxin Zheng, Guolin Ke, Yifei Shen, Jiacheng You, Jiyan He, Shengjie Luo, Chang Liu, Di He, and Tie-Yan Liu. Benchmarking graphormer on large-scale molecular modeling datasets. *arXiv preprint arXiv:2203.04810*, 2022.

[54] Muhammed Shuaibi, Adeesh Kolluru, Abhishek Das, Aditya Grover, Anuroop Sriram, Zachary Ulissi, and C Lawrence Zitnick. Rotation invariant graph neural networks using spin convolutions. *arXiv preprint arXiv:2106.09575*, 2021.

[55] Philipp Thölke and Gianni De Fabritiis. Torchmd-net: equivariant transformers for neural network based molecular potentials. *arXiv preprint arXiv:2202.02541*, 2022.

[56] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

[57] Oliver T Unke and Markus Meuwly. Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges. *Journal of chemical theory and computation*, 15(6):3678–3693, 2019.

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010, 2017.

[59] Limei Wang, Yi Liu, Yuchao Lin, Haoran Liu, and Shuiwang Ji. Comenet: Towards complete and efficient message passing for 3d molecular graphs. *arXiv preprint arXiv:2206.08515*, 2022.

[60] Tian Xie and Jeffrey C Grossman. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical review letters*, 120(14):145301, 2018.

[61] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International Conference on Machine Learning*, pages 10524–10533. PMLR, 2020.

[62] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

[63] Zhao Xu, Youzhi Luo, Xuan Zhang, Xinyi Xu, Yaochen Xie, Meng Liu, Kaleb Dickerson, Cheng Deng, Maho Nakata, and Shuiwang Ji. Molecule3d: A benchmark for predicting 3d geometries from molecular graphs. *arXiv preprint arXiv:2110.01717*, 2021.

[64] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems*, 34:28877–28888, 2021.

[65] Chengxuan Ying, Mingqi Yang, Shuxin Zheng, Guolin Ke, Shengjie Luo, Tianle Cai, Chenglin Wu, Yuxin Wang, Yanming Shen, and Di He. First place solution of kdd cup 2021 & ogb large-scale challenge graph prediction track. *arXiv preprint arXiv:2106.08279*, 2021.

A Implementation Details of GeoMFormer

Layer Normalizations. Being a Transformer-based model, GeoMFormer also adopts the layer normalization (LN) [2] module for training stability. In the invariant stream, the LN module remains unchanged from the standard design [2, 61]. In particular, we specialized the LN module as Equ-LN in the equivariant stream to satisfy the geometric constraints. Formally, given the equivariant representation $\mathbf{z}_i^E \in \mathbb{R}^{3\times d}$ of the atom i, $\text{Equ-LN}(\mathbf{z}_i^E) = \mathbf{U}(\mathbf{z}_i^E - \mu\mathbf{1}^\top) \odot \gamma$, where $\mu = \frac{1}{d}\sum_{k=1}^{d} \mathbf{Z}^E_{[i,:,k]} \in \mathbb{R}^3$, $\gamma \in \mathbb{R}^d$ is a learnable vector, and $\mathbf{U} \in \mathbb{R}^{3\times 3}$ denotes the inverse square root of the covariance matrix, i.e., $\mathbf{U^{-2}} = \frac{\mathbf{(z_i^E - \mu 1^\top)(z_i^E - \mu 1^\top)^\top}}{\mathbf{d}}$.

Structural Encodings. We follow [53] to incorporate the 3D structural encoding, which serves as the bias term in the softmax attention module. In particular, we consider the Euclidean distance $||\mathbf{r}_i - \mathbf{r}_j||$ between atom i and j. The Gaussian Basis Kernel function [49] is used to encode the interatomic distance, i.e., $b^k_{(i,j)} = -\frac{1}{\sqrt{2\pi}|\sigma^k|}\exp(-\frac{1}{2}(\frac{\gamma_{(i,j)}||\mathbf{r}_i - \mathbf{r}_j|| + \beta_{(i,j)} - \mu^k}{|\sigma^k|})^2), k = 1, ..., K$, where K is the number of Gaussian Basis kernels. The 3D structural encoding is obtained by $B_{ij} = \text{GELU}(b_{(i,j)}W_D^1)W_D^2$, where $b_{(i,j)} = [b^1_{(i,j)}; ...; b^K_{(i,j)}]^\top$, $W_D^1 \in \mathbb{R}^{K\times K}$, $W_D^2 \in \mathbb{R}^{K\times 1}$ are learnable parameters. $\gamma_{(i,j)}, \beta_{(i,j)}$ are learnable scalars indexed by the pair of atom types, and μ^k, σ^k are learnable kernel center and learnable scaling factor of the k-th Gaussian Basis Kernel. Denote B as the matrix form of the 3D distance encoding, whose shape is $n \times n$. Then the attention probability is calculated by $\text{softmax}(\frac{QK^\top}{\sqrt{d}} + B)$, where Q and K are the query and key introduced in Section 3.

B Proof of Geometric Constraints

In this section, we provide thorough proof of the aforementioned conditions in Section 4 that satisfy the geometric constraints. For the sake of convenience, we restate the notations and geometric constraints here. Formally, let $V_\mathcal{M}$ denote the space of molecular systems, for each atom i, we define equivariant representation ϕ^E and invariant representation ϕ^I if $\forall\, g = (\mathbf{t}, \mathbf{R}) \in SE(3), \mathcal{M} = (\mathbf{X}, R) \in V_\mathcal{M}$, the following conditions are satisfied:

$$\phi^E : V_\mathcal{M} \to \mathbb{R}^{3\times d}, \qquad \mathbf{R}\phi^E(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^E(\mathbf{X}, \{\mathbf{R}\mathbf{r}_1, ..., \mathbf{R}\mathbf{r}_n\}) \tag{9a}$$

$$\phi^E : V_\mathcal{M} \to \mathbb{R}^{3\times d}, \qquad \phi^E(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^E(\mathbf{X}, \{\mathbf{r}_1 + \mathbf{t}, ..., \mathbf{r}_n + \mathbf{t}\}) \tag{9b}$$

$$\phi^I : V_\mathcal{M} \to \mathbb{R}^{d}, \qquad \phi^I(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^I(\mathbf{X}, \{\mathbf{R}\mathbf{r}_1 + \mathbf{t}, ..., \mathbf{R}\mathbf{r}_n + \mathbf{t}\}) \tag{9c}$$

where $\mathbf{t} \in \mathbb{R}^3, \mathbf{R} \in \mathbb{R}^{3\times 3}, \det(\mathbf{R}) = 1$ and $\mathbf{X} \in \mathbb{R}^{n\times d}$ denotes the atoms with features, $R = \{\mathbf{r}_1, ..., \mathbf{r}_n\}, \mathbf{r}_i \in \mathbb{R}^3$ denotes the cartesian coordinate of atom i. We present the proof of the General Design Philosophy (Section B.1) and our GeoMFormer model (Section B.2) respectively.

B.1 Proof of the General Design Philosophy.

Given invariant and equivariant representations $\mathbf{Z}^{I,0} \in \mathbb{R}^{n\times d}, \mathbf{Z}^{E,0} \in \mathbb{R}^{n\times 3\times d}$ at the input, we prove that the update rules shown in Eqn.(4) satisfy the above constraints in proper conditions. In particular, we first separately study each component of the block, i.e., Inv-Self-Attn, Equ-Self-Attn, Inv-Cross-Attn, Equ-Cross-Attn, and then check the properties of the whole framework.

Invariant Self-Attention. Given invariant representation $\mathbf{Z}^{I,l} \in \mathbb{R}^{n\times d}$, $\mathbf{Q}^{I,l} = \psi_Q^{I,l}(\mathbf{Z}^{I,l}), \mathbf{K}^{I,l} = \psi_K^{I,l}(\mathbf{Z}^{I,l}), \mathbf{V}^{I,l} = \psi_V^{I,l}(\mathbf{Z}^{I,l})$, as stated in Section 4.1, where $\psi^{I,l} : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d}$ is invariant. In this regard, $\forall g = (\mathbf{t}, \mathbf{R}) \in SE(3)$, $\mathbf{Q}^{I,l}, \mathbf{K}^{I,l}, \mathbf{V}^{I,l}$ remain unchanged, which means that $\text{Inv-Self-Attn}(\mathbf{Q}^{I,l}, \mathbf{K}^{I,l}, \mathbf{V}^{I,l})$ also remains unchanged. Then the invariance of the output representations is preserved.

Equivariant Self-Attention. Given equivariant representation $\mathbf{Z}^{E,l} \in \mathbb{R}^{n\times 3\times d}$, $\mathbf{Q}^{E,l} = \psi_Q^{E,l}(\mathbf{Z}^{E,l}), \mathbf{K}^{E,l} = \psi_K^{E,l}(\mathbf{Z}^{E,l}), \mathbf{V}^{E,l} = \psi_V^{E,l}(\mathbf{Z}^{E,l})$, as stated in Section 4.1, where

$\psi^{E,l} : \mathbb{R}^{n\times 3\times d} \rightarrow \mathbb{R}^{n\times 3\times d}$ is equivariant. Besides, the attention score is modified as $\alpha_{ij} = \sum_{k=1}^{d} \mathbf{Q}^{E}{}_{[i,:,k]}\mathbf{K}^{E}{}_{[j,:,k]}{}^{\top}$, where $\mathbf{Q}^{E}{}_{[i,:,k]} \in \mathbb{R}^{3}$ denotes the k-th dimension of the atom i's Query. First, we check the rotation equivariance of the $\mathrm{Equ\text{-}Self\text{-}Attn}$. Given any orthogonal transformation matrix $\mathbf{R} \in \mathbb{R}^{3\times 3}, \det(\mathbf{R}) = 1$, we have $\sum_{k=1}^{d} \mathbf{Q}^{E}{}_{[i,:,k]}\mathbf{R}(\mathbf{K}^{E}{}_{[j,:,k]}\mathbf{R})^{\top} = \sum_{k=1}^{d} \mathbf{Q}^{E}{}_{[i,:,k]}\mathbf{R}\mathbf{R}^{\top}\mathbf{K}^{E}{}_{[j,:,k]}{}^{\top} = \sum_{k=1}^{d} \mathbf{Q}^{E}{}_{[i,:,k]}\mathbf{K}^{E}{}_{[j,:,k]}{}^{\top} = \alpha_{ij}$, which preserves the invariance. As $\psi^{E,l}$ is equivariant, we have $\psi^{E,l}([\mathbf{R}\mathbf{Z}_1^{E,l};,...,;\mathbf{R}\mathbf{Z}_n^{E,l}]) = [\mathbf{R}\psi^{E,l}(\mathbf{Z}^{E,l})_1;,...,;\mathbf{R}\psi^{E,l}(\mathbf{Z}^{E,l})_n]$. Since the output equivariant representation of atom i preserves the equivariance, i.e., $\sum_{j=1}^{n} \frac{\exp(\alpha_{ij})}{\sum_{j'=1}^{n}\exp(\alpha_{ij'})}\mathbf{R}\mathbf{V}_j^{E,l} = \mathbf{R}(\sum_{j=1}^{n} \frac{\exp(\alpha_{ij})}{\sum_{j'=1}^{n}\exp(\alpha_{ij'})}\mathbf{V}_j^{E,l})$, the rotation equivariance is satisfied. Moreover, since the equivariant representation $\mathbf{Z}^{E,l}$ preserves the translation invariance (Eqn.(9b)), the output equivariant representation of $\mathrm{Equ\text{-}Self\text{-}Attn}$ naturally satisfies this constraint.

Cross-Attention modules. As stated in Section 4.1, the Query, Key, and Value of $\mathrm{Inv\text{-}Cross\text{-}Attn}$ are specified as $\mathbf{Q}^{I\_E,l} = \psi_Q^{I,l}(\mathbf{Z}^{I,l}), \mathbf{K}^{I\_E,l} = \psi_K^{I\_E,l}(\mathbf{Z}^{I,l}, \mathbf{Z}^{E,l}), \mathbf{V}^{I\_E,l} = \psi_V^{I\_E,l}(\mathbf{Z}^{I,l}, \mathbf{Z}^{E,l})$, where $\psi^{I,l}, \psi^{I\_E,l}$ are invariant. That is to say, $\forall g = (\mathbf{t}, \mathbf{R}) \in SE(3)$, $\mathbf{Q}^{I\_E,l}, \mathbf{K}^{I\_E,l}, \mathbf{V}^{I\_E,l}$ remain unchanged. Then the invariance of its output representations is preserved as in $\mathrm{Inv\text{-}Self\text{-}Attn}$. On the other hand, the Query, Key, and Value of $\mathrm{Equ\text{-}Cross\text{-}Attn}$ are specified as $\mathbf{Q}^{E\_I,l} = \psi_Q^{E,l}(\mathbf{Z}^{E,l}), \mathbf{K}^{E\_I,l} = \psi_K^{E\_I,l}(\mathbf{Z}^{E,l}, \mathbf{Z}^{I,l}), \mathbf{V}^{E\_I,l} = \psi_V^{E\_I,l}(\mathbf{Z}^{E,l}, \mathbf{Z}^{I,l})$, where $\psi^{E,l}, \psi^{E\_I,l}$ are equivariant, i.e., $\psi^{E\_I,l}([\mathbf{R}\mathbf{Z}_1^{E,l};,...,;\mathbf{R}\mathbf{Z}_n^{E,l}], \mathbf{Z}^{I,l}) = [\mathbf{R}\psi^{E\_I,l}(\mathbf{Z}^{E,l}, \mathbf{Z}^{I,l})_1;,...,;\mathbf{R}\psi^{E,l}(\mathbf{Z}^{E,l}, \mathbf{Z}^{I,l})_n]$ and $\psi^{E,l}([\mathbf{R}\mathbf{Z}_1^{E,l};,...,;\mathbf{R}\mathbf{Z}_n^{E,l}]) = [\mathbf{R}\psi^{E,l}(\mathbf{Z}^{E,l})_1;,...,;\mathbf{R}\psi^{E,l}(\mathbf{Z}^{E,l})_n]$. As stated in $\mathrm{Equ\text{-}Self\text{-}Attn}$, the output equivariant representations of $\mathrm{Equ\text{-}Cross\text{-}Attn}$ preserve the rotation equivariance. Similarly, the translation invariance property is also naturally satisfied.

Feed-Forward Networks. As $\mathrm{Inv\text{-}FFN}$ and $\mathrm{Equ\text{-}FFN}$ satisfy the invariance and equivariance constraints respectively, we can directly obtain that $\forall g = (\mathbf{t}, \mathbf{R}) \in SE(3)$, the output of $\mathrm{Inv\text{-}FFN}$ remains unchanged, and the output of $\mathrm{Equ\text{-}FFN}$ preserves the rotation equivariance, i.e., $\mathrm{Equ\text{-}FFN}([\mathbf{R}\mathbf{Z}_1^{E,l};,...,;\mathbf{R}\mathbf{Z}_n^{E,l}]) = [\mathbf{R}\,\mathrm{Equ\text{-}FFN}(\mathbf{Z}^{E,l})_1;,...,;\mathbf{R}\,\mathrm{Equ\text{-}FFN}(\mathbf{Z}^{E,l})_n]$. The translation invariance is also naturally preserved by $\mathrm{Equ\text{-}FFN}$.

With the above analysis, the update rules stated in Eqn.(4) satisfy the geometric constraints (Eqn.(9a), Eqn.(9b) and Eqn.(9c)). As our model is composed of stacked blocks, the invariant and equivariant output representations of the whole model also preserve the constraints.

B.2 Proof of the GeoMFormer

Next, we provide proof of the instantiation of our GeoMFormer in Section 4.2 that satisfies the geometric constraints. Similarly, we separately check the properties of each component as our GeoMFormer is composed of stacked GeoMFormer blocks. Once the constraints are satisfied by each component, the output invariant and equivariant representations of the whole model naturally satisfy the geometric constraints (Eqn.(9a), Eqn.(9b) and Eqn.(9c)).

Input layer. As stated in Section 4.2, the invariant representation at the input is set as $\mathbf{Z}^{I,0} = \mathbf{X}$, where $\mathbf{X}_i \in \mathbb{R}^d$ is a learnable embedding vector indexed by the atom i's type. Since $\mathbf{Z}^{I,0}$ does not contain any information from $R = \{\mathbf{r}_1, ..., \mathbf{r}_n\}$, it naturally satisifies the invariance constraint (Eqn.(9c)). The equivariant representation at the input is set as $\mathbf{Z}_i^{E,0} = \hat{\mathbf{r}}'_i g(||\mathbf{r}'_i||)^{\top} \in \mathbb{R}^{3\times d}$, where $\mathbf{r}'_i$ denotes the mean-centered position of atom i, i.e., $\mathbf{r}'_i = \mathbf{r}_i - \frac{1}{n}\sum_{k=1}^{n}\mathbf{r}_k$, $\hat{\mathbf{r}}'_i = \frac{\mathbf{r}'_i}{||\mathbf{r}'_i||}$, and $g : \mathbb{R} \rightarrow \mathbb{R}^d$ is instantiated by the Gaussian Basis Kernel function. First, the translation invariance constraint (Eqn.(9b)) is satisfied. Given any translation vector $\mathbf{t} \in \mathbb{R}^3$, $\mathbf{r}_i + \mathbf{t} - \frac{1}{n}\sum_{k=1}^{n}(\mathbf{r}_k + \mathbf{t}) = \mathbf{r}_i - \frac{1}{n}\sum_{k=1}^{n}\mathbf{r}_k$, and $\mathbf{Z}_i^{E,0}$ remains unchanged. Second, the rotation equivariance (Eqn.(9a)) is also preserved. Given any orthogonal transformation matrix $\mathbf{R} \in \mathbb{R}^{3\times 3}, \det(\mathbf{R}) = 1$, we have $||\mathbf{R}\mathbf{r}'_i|| = ||\mathbf{r}'_i||$. With $\mathbf{R}\mathbf{r}_i$ as the input, we have $\mathbf{R}\mathbf{r}_i - \frac{1}{n}\sum_{k=1}^{n}\mathbf{R}\mathbf{r}_k = \mathbf{R}(\mathbf{r}_i - \frac{1}{n}\sum_{k=1}^{n}\mathbf{r}_k) = \mathbf{R}\mathbf{r}'_i$ and $g(||\mathbf{R}\mathbf{r}'_i||) = g(||\mathbf{r}'_i||)$, which means that the rotation equivariance constraint is satisfied.

Self-Attention modules. For $\mathrm{Inv\text{-}Self\text{-}Attn}$ and $\mathrm{Equ\text{-}Self\text{-}Attn}$, we use the linear function to implement both ψ^I and ψ^E, i.e., $\mathbf{Q}^I = \psi^I_Q(\mathbf{Z}^I) = \mathbf{Z}^I W^I_Q, \mathbf{K}^I = \psi^I_K(\mathbf{Z}^I) = \mathbf{Z}^I W^I_K, \mathbf{V}^I = \psi^I_V(\mathbf{Z}^I) = \mathbf{Z}^I W^I_V$ and $\mathbf{Q}^E = \psi^E_Q(\mathbf{Z}^E) = \mathbf{Z}^E W^E_Q, \mathbf{K}^E = \psi^E_K(\mathbf{Z}^E) = \mathbf{Z}^E W^E_K, \mathbf{V}^E = \psi^E_V(\mathbf{Z}^E) = \mathbf{Z}^E W^E_V$. It is straightforward that the conditions mentioned in Section B.1 are satisfied. The linear function keeps the invariance of $\mathbf{Z}^I$ (Eqn.(9c)) and the rotation equivariance of $\mathbf{Z}^E$ (Eqn.(9a)), e.g., $\forall \mathbf{R} \in \mathbb{R}^{3\times3}, \det(\mathbf{R}) = 1, (\mathbf{R}\mathbf{Z}^E_i)W^E_Q = \mathbf{R}(\mathbf{Z}^E_i W^E_Q) = \mathbf{R}\mathbf{Z}^E_i$. Note that the translation invariance of $\mathbf{Z}^E$ (Eqn.(9b)) is not changed by the linear function.

Cross-Attention modules. For $\mathrm{Inv\text{-}Cross\text{-}Attn}$, we use the linear function to implement ψ^I_Q, which satisfies the constraints as previously stated. Besides, we instantiate $\mathbf{K}^{I\_E}$ and $\mathbf{V}^{I\_E}$ as $\mathbf{K}^{I\_E} = \psi^{I\_E}_K(\mathbf{Z}^I, \mathbf{Z}^E) =< \mathbf{Z}^E W^{I\_E}_{K,1}, \mathbf{Z}^E W^{I\_E}_{K,2} >, \mathbf{V}^{I\_E} = \psi^{I\_E}_V(\mathbf{Z}^I, \mathbf{Z}^E) =< \mathbf{Z}^E W^{I\_E}_{V,1}, \mathbf{Z}^E W^{I\_E}_{V,2} >$. Here we prove that such instantiation preserve the invariance. First, given any orthogonal transformation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}, \det(\mathbf{R}) = 1$, we have $< ([\mathbf{R}\mathbf{Z}^E_1; ...; \mathbf{R}\mathbf{Z}^E_n])W^{I\_E}_{K,1}, ([\mathbf{R}\mathbf{Z}^E_1; ...; \mathbf{R}\mathbf{Z}^E_n])W^{I\_E}_{K,2} >=< \mathbf{Z}^E W^{I\_E}_{K,1}, \mathbf{Z}^E W^{I\_E}_{K,2} >$. The reason is that given $X, Y \in \mathbb{R}^{n\times3\times d}, Z =< X, Y >\in \mathbb{R}^{n\times d}$, where $Z_{[i,k]} = {X_{[i,:,k]}}^\top Y_{[i,:,k]} = {X_{[i,:,k]}}^\top \mathbf{R}^\top \mathbf{R} Y_{[i,:,k]} = (\mathbf{R}X_{[i,:,k]})^\top(\mathbf{R}Y_{[i,:,k]})$. The translation invariance of $\mathbf{Z}^E$ is also preserved.

For $\mathrm{Equ\text{-}Cross\text{-}Attn}$, we also use the linear function to implement ψ^E_Q, which satisfies the constraints as previously stated. Besides, we instantiate $\mathbf{K}^{E\_I}$ and $\mathbf{V}^{E\_I}$ as $\mathbf{K}^{E\_I} = \psi^{E\_I}_K(\mathbf{Z}^E, \mathbf{Z}^I) = \mathbf{Z}^E W^{E\_I}_{K,1} \odot \mathbf{Z}^I W^{E\_I}_{K,2}, \mathbf{V}^{E\_I} = \psi^{E\_I}_V(\mathbf{Z}^E, \mathbf{Z}^I) = \mathbf{Z}^E W^{E\_I}_{V,1} \odot \mathbf{Z}^I W^{E\_I}_{V,2}$. First, given any orthogonal transformation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$, we have $([\mathbf{R}\mathbf{Z}^E_1; ...; \mathbf{R}\mathbf{Z}^E_n])W^{E\_I}_{K,1} \odot \mathbf{Z}^I W^{E\_I}_{K,2} = [\mathbf{R}(\mathbf{Z}^E W^{E\_I}_{K,1} \odot \mathbf{Z}^I W^{E\_I}_{K,2})_1; ...; \mathbf{R}(\mathbf{Z}^E W^{E\_I}_{K,1} \odot \mathbf{Z}^I W^{E\_I}_{K,2})_n]$, which preserves the rotation equivariance. The reason lies in that given $X \in \mathbb{R}^{n\times3\times d}, Y \in \mathbb{R}^{n\times d}$, $Z_i = \mathbf{R}X_i \odot Y_i \in \mathbb{R}^{3\times d}$, where $Z_{[i,:,k]} = (\mathbf{R}X_{[i,:,k]}) \cdot Y_{[i,k]} = \mathbf{R}(X_{[i,:,k]} \cdot Y_{[i,k]})$. Additionally, the translation invariance of both $\mathbf{K}^{E\_I}$ and $\mathbf{V}^{E\_I}$ is preserved because of the translation invariance of $\mathbf{Z}^E$ and $\mathbf{Z}^I$. In this way, the instantiations of cross-attention modules satisfy the geometric constraints.

Feed-Forward Networks. For $\text{Inv-FFN}({\mathbf{Z}''}^I) = \text{GELU}({\mathbf{Z}''}^I W^I_1)W^I_2$, the invariance constraint (Eqn. 9c) is naturally preserved. For $\text{Equ-FFN}({\mathbf{Z}''}^E) = ({\mathbf{Z}''}^E W^E_1 \odot \text{GELU}({\mathbf{Z}''}^I W^I_2))W^E_3$, the rotation equivariance constraint is also similarly preserved as in $\mathrm{Equ\text{-}Cross\text{-}Attn}$. Besides, the translation invariance of $\text{Equ-FFN}({\mathbf{Z}''}^E)$ is also preserved with the property of ${\mathbf{Z}''}^E$ and ${\mathbf{Z}''}^I$.

Layer Normalizations. As introduced in Section A, we use the layer normalization modules for both invariant and equivariant streams. For the invariant stream, the layer normalization remains unchanged, and the invariance constraint is naturally preserved. For the equivariant stream, given the equivariant representation $\mathbf{z}^E_i \in \mathbb{R}^{3\times d}$ of the atom i, $\text{Equ-LN}(\mathbf{z}^E_i) = \mathbf{U}(\mathbf{z}^E_i - \mu\mathbf{1}^\top) \odot \gamma$, where $\mu = \frac{1}{d}\sum_{k=1}^{d} \mathbf{Z}^E_{[i,:,k]} \in \mathbb{R}^3$, $\gamma \in \mathbb{R}^d$ is a learnable vector, and $\mathbf{U} \in \mathbb{R}^{3\times3}$ denotes the inverse square root of the covariance matrix, i.e., $\mathbf{U}^{-2} = \frac{(\mathbf{z}^E_i - \mu\mathbf{1}^\top)(\mathbf{z}^E_i - \mu\mathbf{1}^\top)^\top}{d}$. First, given any orthogonal transformation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}, \det(\mathbf{R}) = 1$, $\frac{(\mathbf{R}\mathbf{z}^E_i - \mathbf{R}\mu\mathbf{1}^\top)(\mathbf{R}\mathbf{z}^E_i - \mathbf{R}\mu\mathbf{1}^\top)^\top}{d} = \frac{(\mathbf{R}\mathbf{z}^E_i - \mathbf{R}\mu\mathbf{1}^\top)(\mathbf{R}\mathbf{z}^E_i - \mathbf{R}\mu\mathbf{1}^\top)^\top}{d} = \mathbf{R}\frac{(\mathbf{z}^E_i - \mu\mathbf{1}^\top)(\mathbf{z}^E_i - \mu\mathbf{1}^\top)^\top}{d}\mathbf{R}^\top = \mathbf{R}\mathbf{U}^{-2}\mathbf{R}^\top = \mathbf{R}\mathbf{U}^{-1}\mathbf{R}^\top\mathbf{R}\mathbf{U}^{-1}\mathbf{R}^\top = (\mathbf{R}\mathbf{U}\mathbf{R}^\top)^{-2}$, then we have $\text{Equ-LN}(\mathbf{R}\mathbf{z}^E_i) = \mathbf{R}\mathbf{U}\mathbf{R}^\top(\mathbf{R}\mathbf{z}^E_i - \mathbf{R}\mu\mathbf{1}^\top) \odot \gamma = \mathbf{R}(\mathbf{U}(\mathbf{z}^E_i - \mu\mathbf{1}^\top)) = \mathbf{R}\,\text{Equ-LN}(\mathbf{z}^E_i)$, which preserves the rotation equivariance (Eqn.(9a)). The translation invariance of $\mathbf{Z}^E$ is also preserved.

Structural Encodings. As introduced in Section A, the structural encodings serve as the bias term in the softmax attention module. Since only the relative distance $||\mathbf{r}_i - \mathbf{r}_j||, \forall i, j \in [n]$ is used, the invariance constraint is preserved, i.e., given $\forall g = (\mathbf{t}, \mathbf{R}) \in SE(3)$, $||\mathbf{R}\mathbf{r}_i + \mathbf{t} - \mathbf{R}\mathbf{r}_j + \mathbf{t}|| = ||\mathbf{r}_i - \mathbf{r}_j||$.

C Discussions

C.1 Connections to previous approaches

In this section, we present a detailed discussion of how previous models (PaiNN [50] and TorchMD-Net [55]) can be viewed as special instantiations by extending the design philosophy described in Section 4.1. Without loss of generality, we omit the cutoff conditions used in these works for readability, which can be naturally included in our framework.

PaiNN [50]. Both invariant representations $\mathbf{Z}^I = [\mathbf{z}_1^{I^\top}; ...; \mathbf{z}_n^{I^\top}] \in \mathbb{R}^{n\times d}$ and equivariant representations $\mathbf{Z}^E = [\mathbf{z}_1^E; ...; \mathbf{z}_n^E] \in \mathbb{R}^{n\times 3\times d}$ are maintained in PaiNN, where $\mathbf{z}_i^I \in \mathbb{R}^d$ and $\mathbf{z}_i^E \in \mathbb{R}^{3\times d}$ are the invariant and equivariant representations for atom i, respectively. In each layer, the representations are updated as follows:

$$\begin{aligned} \mathbf{Z}'^{I,l} &= \mathbf{Z}^{I,l} \ + \text{Message-Block-Inv}(\mathbf{Z}^{I,l}) \\ \mathbf{Z}'^{E,l} &= \mathbf{Z}^{E,l} \ + \text{Message-Block-Equ}(\mathbf{Z}^{I,l}, \mathbf{Z}^{E,l}) \\ \mathbf{Z}^{I,l+1} &= \mathbf{Z}'^{I,l} \ + \text{Update-Block-Inv}(\mathbf{Z}'^{I,l}, \mathbf{Z}'^{E,l}) \\ \mathbf{Z}^{E,l+1} &= \mathbf{Z}'^{E,l} \ + \text{Update-Block-Equ}(\mathbf{Z}'^{I,l}, \mathbf{Z}'^{E,l}) \end{aligned} \tag{10}$$

In the message block, the invariant and equivariant representations are updated in the following manner. For brevity, we omit the layer index l.

$$\begin{aligned} \text{Message-Block-Inv}(\mathbf{z}_i^I) &= \sum_j \phi_s(\mathbf{z}_j^I) \circ \mathcal{W}_s(||\mathbf{r}_i - \mathbf{r}_j||) \\ \text{Message-Block-Equ}(\mathbf{z}_i^I, \mathbf{z}_i^E) &= \sum_j \mathbf{z}_j^E \odot \Big(\phi_{vv}(\mathbf{z}_j^I) \circ \mathcal{W}_{vv}(||\mathbf{r}_i - \mathbf{r}_j||)\Big) \\ &+ \frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||} \Big(\phi_{vs}(\mathbf{z}_j^I) \circ \mathcal{W}_{vs}^{'}(||\mathbf{r}_i - \mathbf{r}_j||)\Big)^\top \end{aligned} \tag{11}$$

The scalar product $\odot$ is defined the same way as in Section 4.2, i.e., given $x \in \mathbb{R}^{3\times d}, y \in \mathbb{R}^d, z = x \odot y \in \mathbb{R}^{3\times d}$, where $z_{[i,j]} = x_{[i,k]} \cdot y_{[k]}$. $\circ$ denotes the element-wise product, $\phi_s, \phi_{vv}, \phi_{vs} : \mathbb{R}^d \to \mathbb{R}^d$ are all 2-layer MLP with the SiLU activation, $\mathcal{W}_s, \mathcal{W}_{vv}, \mathcal{W}'_{vs} : \mathbb{R} \to \mathbb{R}^d$ are instantiated by learnable radial basis functions. $\frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||} \in \mathbb{R}^3$ denotes the relative direction between atom i's and j's positions.

In the update block, the invariant and equivariant representations are updated in the following manner:

$$\begin{aligned} \text{Update-Block-Inv}(\mathbf{z}_i^I, \mathbf{z}_i^E) &= \mathbf{a}_{ss}(\mathbf{z}_i^I, ||\mathbf{z}_i^E \mathbf{V}||) + \mathbf{a}_{sv}(\mathbf{z}_i^I, ||\mathbf{z}_i^E \mathbf{V}||) \circ < \mathbf{z}_i^E \mathbf{U}, \mathbf{z}_i^E \mathbf{V} > \\ \text{Update-Block-Equ}(\mathbf{z}_i^I, \mathbf{z}_i^E) &= \mathbf{a}_{vv}(\mathbf{z}_i^I, ||\mathbf{z}_i^E \mathbf{V}||) \odot (\mathbf{z}_i^E \mathbf{U}) \end{aligned}$$

$\mathbf{V}, \mathbf{U} \in \mathbb{R}^{d\times d}$ are learnable parameters. $< \cdot, \cdot >$ is defined the same way as in Section 4.2, i.e., given $x, y \in \mathbb{R}^{3\times d}, z =< x, y >\in \mathbb{R}^d$, where $z_{[k]} = x_{[:,k]}{}^\top y_{[:,k]}$. Norm $|| \cdot || : \mathbb{R}^{3\times d} \to \mathbb{R}^d$ is calculated along the spatial dimension, i.e., $|| \cdot || =< \cdot, \cdot >$. $\circ$ denotes the element-wise product. $\odot$ is also defined the same as in Section 4.2. $\mathbf{a}(\cdot, \cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ first concatenates the two inputs along the feature dimension and then apply a 2-layer MLP with SiLU activation.

We prove that both the invariant and equivariant message blocks can be viewed as special instances by extending the invariant self-attention module and the equivariant cross-attention module of our framework respectively. In particular, we extend $\psi_V^I, \psi_V^{E\_I}$ introduced in the Section 4.1 to be query-dependent, i.e., $\psi_V^{I,i}, \psi_V^{E\_I,i}$ that depends on the atom i's representations. Concretely, in the invariant self-attention module, we set $\psi_V^{I,\,i}(\mathbf{z}_j^I) = \phi_s(\mathbf{z}_j^I) \odot \mathcal{W}_s(||\mathbf{r}_i - \mathbf{r}_j||)$. Similarly, in the equivariant cross-attention module, we set $\psi_V^{E\_I,\,i}(\mathbf{z}_j^I, \mathbf{z}_j^E) = \mathbf{z}_j^E \odot \phi_{vv}(\mathbf{z}_j^I) \cdot \mathcal{W}_{vv}(||\mathbf{r}_i - \mathbf{r}_j||) + \phi_{vs}(\mathbf{z}_j^I) \cdot \mathcal{W}_{vs}^{'} \frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||}$. In such way, the invariant self-attention module and the equivariant cross-attention module can express the invariant and equivariant message blocks respectively, e.g., the parameters to transform Query and Key are trained/initialized to zero, and the number of atoms can be equipped by initialization, which is necessary to express the sum operator by using the attention as shown in [64].

Moreover, we prove that the update blocks can also be viewed as special instances by extending the FFN blocks in our framework. In particular, we set $\text{Inv-FFN}(\mathbf{z}_i^I) = \mathbf{a}_{ss}(\mathbf{z}_i^I, ||\mathbf{z}_i^E \mathbf{V}||) + \mathbf{a}_{sv}(\mathbf{z}_i^I, ||\mathbf{z}_i^E \mathbf{V}||) < \mathbf{z}_i^E \mathbf{U}, \mathbf{z}_i^E \mathbf{V} >$ and $\text{Equ-FFN}(\mathbf{z}_i^E) = \mathbf{a}_{vv}(\mathbf{z}_i^I, ||\mathbf{z}_i^E \mathbf{V}||) \left(\mathbf{z}_i^E \mathbf{U}\right)$, then both Inv-FFN and Equ-FFN can express the update blocks. Note that the parameters of the remaining blocks $(\text{Inv-Cross-Attn}, \text{Equ-Self-Attn})$ can be trained/initialized to be zero. In such way, the PaiNN model can be instantiated through our design philosophy introduced in Section 4.1.

TorchMD-Net [55]. Similarly to PaiNN, both invariant representations $\mathbf{Z}^I = [\mathbf{z}_1^{I^\top}; ...; \mathbf{z}_n^{I^\top}] \in \mathbb{R}^{n\times d}$ and equivariant representations $\mathbf{Z}^E = [\mathbf{z}_1^E; ...; \mathbf{z}_n^E] \in \mathbb{R}^{n\times 3\times d}$ are maintained in TorchMD-

Net, where $\mathbf{z}_i^I \in \mathbb{R}^d$ and $\mathbf{z}_i^E \in \mathbb{R}^{3\times d}$ are the invariant and equivariant representations for atom i, respectively. In each layer, the representations are updated as follows:

$$\begin{aligned}
\mathbf{Z}'^{I,l} &= \mathbf{Z}^{I,l} + \text{TorchMD-Inv-Block-1}(\mathbf{Z}^{I,l}) \\
\mathbf{Z}^{I,l+1} &= \mathbf{Z}'^{I,l} + \text{TorchMD-Inv-Block-2}(\mathbf{Z}'^{I,l}, \mathbf{Z}^{E,l}) \\
\mathbf{Z}^{E,l+1} &= \mathbf{Z}^{E,l} + \text{TorchMD-Equ-Block}(\mathbf{Z}^{I,l}, \mathbf{Z}^{E,l})
\end{aligned} \tag{12}$$

The invariant representations in TorchMD-Inv-Block-1 and TorchMD-Inv-Block-2 are updated as follows. For brevity, we omit the layer index l.

$$\begin{aligned}
\mathbf{Q}_i = W^Q\mathbf{z}_i^I, \mathbf{K}_j &= W^K\mathbf{z}_j^I, \mathbf{V}_j^{(1)} = W^{V(1)}\mathbf{z}_j^I \\
\alpha_{ij} &= \text{SiLU}\left(\mathbf{Q}_i^\top\left(\mathbf{K}_j \circ \mathbf{D}_{ij}^K\right)\right) \\
\text{TorchMD-Inv-Block-1}(\mathbf{z}_i^I) &= O_1\left(\sum_j \alpha_{ij}\cdot \mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V(1)}\right) \\
\text{TorchMD-Inv-Block-2}(\mathbf{z}_i^I, \mathbf{z}_i^E) &= O_2\left(\sum_j \alpha_{ij}\cdot \mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V(1)}\right) \circ < \mathbf{z}_i^E\mathbf{U}_1, \mathbf{z}_i^E\mathbf{U}_2 >
\end{aligned} \tag{13}$$

$\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^{V(1)}, \mathbf{U}_1, \mathbf{U}_2 \in \mathbb{R}^{d\times d}$ are learnable parameters. $\circ$ denotes the element-wise product. $\mathbf{D}_{ij}^K, \mathbf{D}_{ij}^{V(1)} : \mathbb{R} \to \mathbb{R}^d$ takes $||\mathbf{r}_i - \mathbf{r}_j||$ as input and uses radial basis functions followed by a non-linear activation to transform it. $O_1, O_2 : \mathbb{R}^d \to \mathbb{R}^d$ are learnable linear transformations. $< \cdot, \cdot >$ is defined the same way as in Section 4.2, i.e., given $x, y \in \mathbb{R}^{3\times d}, z =< x, y >\in \mathbb{R}^d$, where $z_{[k]} = {x_{[:,k]}}^\top y_{[:,k]}$. On the other hand, the equivariant representations are updated as follows:

$$\mathbf{V}_j^{(2)} = \mathbf{W}^{V(2)}\mathbf{z}_j^I, \mathbf{V}_j^{(3)} = \mathbf{W}^{V(3)}\mathbf{z}_j^I$$

$$\begin{aligned}
\text{TorchMD-Equ-Block}(\mathbf{z}_i^I, \mathbf{z}_i^E) = &\sum_j \left((\mathbf{V}_j^{(2)} \circ \mathbf{D}_{ij}^{V(2)}) \odot \mathbf{z}_j^E + \frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||}(\mathbf{V}_j^{(3)} \circ \mathbf{D}_{ij}^{V(3)})^\top\right) \\
&+ O_3\left(\sum_j \alpha_{ij}\cdot \mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V(1)}\right) \odot \mathbf{z}_i^E\mathbf{U}_3
\end{aligned} \tag{14}$$

$\mathbf{W}^{V(2)}, \mathbf{W}^{V(3)}, \mathbf{U}_3 \in \mathbb{R}^{d\times d}$ are learnable parameters. $\circ$ denotes the element-wise product. $\odot$ is defined the same way as in Section 4.2, i.e., given $x \in \mathbb{R}^{3\times d}, y \in \mathbb{R}^d, z = x \odot y \in \mathbb{R}^{3\times d}$, where $z_{[i,j]} = x_{[i,k]} \cdot y_{[k]}$. $\mathbf{D}_{ij}^{V^{(2)}}, \mathbf{D}_{ij}^{V^{(3)}} : \mathbb{R} \to \mathbb{R}^d$ takes $||\mathbf{r}_i - \mathbf{r}_j||$ as input and use radial basis functions followed by a non-linear activation to transform it. $O_3 : \mathbb{R}^d \to \mathbb{R}^d$ is a learnable linear transformation. $\frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||} \in \mathbb{R}^3$ denotes the relative direction between atom i's and j's positions.

We prove that the TorchMD-Inv-Block-1 and TorchMD-Inv-Block-2 can be viewed as special instances by extending the invariant self-attention module and invariant cross-attention module of our framework respectively. Concretely, in the invariant self-attention module, we set $\psi_Q^I(\mathbf{z}_i^I) = W^Q\mathbf{z}_i^I, \psi_K^{I,\,i}(\mathbf{z}_j^I) = W^K\mathbf{z}_j^I \circ \mathbf{D}_{ij}^K, \psi_V^{I,\,i}(\mathbf{z}_j^I) = O_1\left(W^{V^{(1)}}\mathbf{z}_j^I \circ \mathbf{D}_{ij}^{V^{(1)}}\right)$ and use SiLU instead of Softmax for calculating attention probability. By rewriting TorchMD-Inv-Block-1 in the equivalent form TorchMD-Inv-Block-1$(\mathbf{z}_i^I) = \sum_j \alpha_{ij}\cdot O_1\left(\mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V^{(1)}}\right)$, the invariant self-attention module can express it by equipping the number of atoms for expressing the sum operation using the attention.

In the invariant cross-attention module, we set $\psi_Q^I(\mathbf{z}_i^I) = W^Q\mathbf{z}_i^I, \psi_K^{I\_E,\,i}(\mathbf{z}_j^I, \mathbf{z}_j^E) = W^K\mathbf{z}_j^I \circ \mathbf{D}_{ij}^K, \psi_V^{I\_E,\,i}(\mathbf{z}_j^I, \mathbf{z}_j^E) = O_2\left(W^{V^{(1)}}\mathbf{z}_j^I \circ \mathbf{D}_{ij}^{V^{(1)}}\right) \circ < U_1\mathbf{z}_i^E, U_2\mathbf{z}_i^E >$, and use SiLU instead of Softmax for calculating attention probability. By rewriting TorchMD-Inv-Block-2 in the equivalent form TorchMD-Inv-Block-2$(\mathbf{z}_i^I, \mathbf{z}_i^E) = \sum_j \alpha_{ij} \cdot O_2\left(\mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V^{(1)}}\right) \circ < U_1\mathbf{z}_i^E, U_2\mathbf{z}_i^E >$, the invariant cross-attention module can express it by equipping the number of atoms.

Moreover, we prove that the TorchMD-Equ-Block can be viewed as a special instance by extending the equivariant cross-attention module of our framework. In particular, we set $\psi_V^{E\_I,\,i}(\mathbf{z}_j^I, \mathbf{z}_j^E) = (W^{V^{(2)}}\mathbf{z}_j^I \circ \mathbf{D}_{ij}^{V^{(2)}}) \odot \mathbf{z}_j^E + \frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||}(W^{V^{(3)}}\mathbf{z}_j^I \circ \mathbf{D}_{ij}^{V^{(3)}})^\top + \alpha_{ij}\cdot O_3\left(W^{V^{(1)}}\mathbf{z}_j^I \circ \mathbf{D}_{ij}^{V^{(1)}}\right) \odot U_3\mathbf{z}_i^E$.

By rewriting TorchMD-Equ-Block in the equivalent form $\mathrm{TorchMD\text{-}Equ\text{-}Block}(\mathbf{z}_i^I, \mathbf{z}_i^E) = \sum_j \left((\mathbf{V}_j^{(2)} \circ \mathbf{D}_{ij}^{V^{(2)}}) \odot \mathbf{z}_j^E + \frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||} (\mathbf{V}_j^{(3)} \circ \mathbf{D}_{ij}^{V^{(3)}})^\top \right) + \sum_j \alpha_{ij} \cdot O_3 \left(\mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V^{(1)}} \right) \odot U_3 \mathbf{z}_i^E = \sum_j \left((\mathbf{V}_j^{(2)} \circ \mathbf{D}_{ij}^{V^{(2)}}) \odot \mathbf{z}_j^E + \frac{\mathbf{r}_i - \mathbf{r}_j}{||\mathbf{r}_i - \mathbf{r}_j||} (\mathbf{V}_j^{(3)} \circ \mathbf{D}_{ij}^{V^{(3)}})^\top + \alpha_{ij} \cdot O_3 \left(\mathbf{V}_j^{(1)} \circ \mathbf{D}_{ij}^{V^{(1)}} \right) \odot U_3 \mathbf{z}_i^E \right)$, it is straightforward that the equivariant cross-attention module can express the TorchMD-Equ-Block, e.g., the parameters to transform Query and Key are trained/initialized to zero, and the number of atoms can be equipped by initialization. Note that the parameters of the remaining blocks $(\mathrm{Equ\text{-}Self\text{-}Attn}, \mathrm{Inv\text{-}FFN}, \mathrm{Equ\text{-}FFN})$ can be trained/initialized to be zero. In such ways, the TorchMD-Net model can be instantiated through our design philosophy introduced in Section 4.1.

C.2 Extension to other geometric constraints

In this subsection, we showcase how to extend our framework to encode other geometric constraints. In particular, we consider the $E(3)$ group, which comprises translation, rotation and reflection. Formally, let $V_{\mathcal{M}}$ denote the space of molecular systems, for each atom i, we define equivariant representation ϕ^E and invariant representation ϕ^I if $\forall\, g = (\mathbf{t}, \mathbf{R}) \in E(3), \mathcal{M} = (\mathbf{X}, R) \in V_{\mathcal{M}}$, the following conditions are satisfied:

$$\phi^E : V_{\mathcal{M}} \to \mathbb{R}^{3 \times d}, \qquad \mathbf{R}\phi^E(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) + \mathbf{t}\mathbf{1}^\top = \phi^E(\mathbf{X}, \{\mathbf{R}\mathbf{r}_1 + \mathbf{t}, ..., \mathbf{R}\mathbf{r}_n + \mathbf{t}\}) \quad (15a)$$

$$\phi^I : V_{\mathcal{M}} \to \mathbb{R}^{d}, \qquad \phi^I(\mathbf{X}, \{\mathbf{r}_1, ..., \mathbf{r}_n\}) = \phi^I(\mathbf{X}, \{\mathbf{R}\mathbf{r}_1 + \mathbf{t}, ..., \mathbf{R}\mathbf{r}_n + \mathbf{t}\}) \quad (15b)$$

where $\mathbf{t} \in \mathbb{R}^3$ is a translation vector, $\mathbf{R} \in \mathbb{R}^{3\times 3}, \det(\mathbf{R}) = \pm 1$ is an orthogonal transformation matrix and $\mathbf{X} \in \mathbb{R}^{n \times d}$ denotes the atoms with features, $R = \{\mathbf{r}_1, ..., \mathbf{r}_n\}, \mathbf{r}_i \in \mathbb{R}^3$ denotes the cartesian coordinate of atom i. In particular, the additional requirement is to encode the translation and reflection equivariance of the equivariant representations, which can be achieved by modifying the conditions of our framework (Eqn.(4)).

With the invariant representation $\mathbf{Z}^I$ and the equivariant representation $\mathbf{Z}^E$ that satisfy the constraints (Eqn.(15a) and Eqn.(15b)), we separately redefine the conditions of each component. It is worth noting that the reflection invariance is directly satisfied ($\mathbf{R}\mathbf{R}^\top = \mathbf{R}^\top\mathbf{R} = \mathbf{I}$) from the analysis in Section B.1 and Section B.2, which is required in (1) the calculation of attention probability in $\mathrm{Equ\text{-}Self\text{-}Attn}, \mathrm{Equ\text{-}Cross\text{-}Attn}$; (2) the calculation of $\mathbf{K}^{I\_E}$ and $\mathbf{V}^{I\_E}$. Thus, we only need to encode the translation equivariance constraint. Given the update rules (Eqn.(4)), it can be achieved by simply setting each component $(\mathrm{Inv\text{-}Self\text{-}Attn}, \mathrm{Inv\text{-}Cross\text{-}Attn}, \mathrm{Equ\text{-}Self\text{-}Attn}, \mathrm{Equ\text{-}Cross\text{-}Attn}, \mathrm{Inv\text{-}FFN}, \mathrm{Equ\text{-}FFN})$ to be translation-invariant. In this way, the output equivariant representation can preserve the equivariance to the $E(3)$ group. We extend our framework to achieve this goal, which is introduced below:

Self-Attention modules. For $\mathrm{Inv\text{-}Self\text{-}Attn}$, the condition remains unchanged. For $\mathrm{Equ\text{-}Self\text{-}Attn}$, the additional condition is that ψ^E should keep the translation invariance. Here we give a simple instantiation: $\mathbf{Q}^E = \psi_Q^E(\mathbf{Z}^E) = (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_Q^E, \mathbf{K}^E = \psi_K^E(\mathbf{Z}^E) = (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_K^E, \mathbf{V}^E = \psi_V^E(\mathbf{Z}^E) = (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_V^E$, where $\mu_{\mathbf{Z}^E, i} = \frac{1}{d}\sum_{k=1}^n \mathbf{Z}^E{}_{[i,:,k]}\mathbf{1}^\top$.

Cross-Attention modules. For $\mathrm{Inv\text{-}Cross\text{-}Attn}$, the condition for ψ^I remains unchanged, while ψ^{I_E} should keep the translation invariance. For $\mathrm{Equ\text{-}Cross\text{-}Attn}$, both ψ^E and ψ^{E_I} are required to be translation-invariant. Here we give an instantiation: $\mathbf{Q}^E = \psi_Q^E(\mathbf{Z}^E) = (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_Q^E$, and

$$\begin{aligned} \mathbf{K}^{I\_E} &=< (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_{K,1}^{I\_E}, (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_{K,2}^{I\_E} >, \quad & \mathbf{V}^{I\_E} &=< (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_{V,1}^{I\_E}, (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_{V,2}^{I\_E} > \\ \mathbf{K}^{E\_I} &= (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_{K,1}^{E\_I} \odot \mathbf{Z}^I W_{K,2}^{E\_I}, & \mathbf{V}^{E\_I} &= (\mathbf{Z}^E - \mu_{\mathbf{Z}^E})W_{V,1}^{E\_I} \odot \mathbf{Z}^I W_{V,2}^{E\_I} \end{aligned} \quad (16)$$

Feed-Forward Networks. Similarly, the condition for $\mathrm{Inv\text{-}FFN}$ remains unchanged. For $\mathrm{Equ\text{-}FFN}$, it also should keep the translation invariance, e.g., $\mathrm{Equ\text{-}FFN}(\mathbf{Z''}^E) = ((\mathbf{Z''}^E - \mu_{\mathbf{Z}^E})W_1^E \odot \mathrm{GELU}(\mathbf{Z''}^I W_2^I))W_3^E$.

Remark. With the above additional conditions, our framework can additionally be extended to encode geometric constraints towards $E(3)$ group. Note that the design of the input layer should also encode the constraints (Eqn.(15a) and Eqn.(15b)). For example, the invariant representation remains unchanged as $\mathbf{Z}^{I,0} = \mathbf{X}$. while the equivariant representation can be directly set as $\mathbf{Z}_i^{E,0} = \mathbf{r}_i$. In this way, the geometric constraints are well satisfied.

Table 6: Results on Molecule3D for both random and scaffold splits. Bold values indicate the best.

	MAE ↓	
Model	Random	Scaffold
GIN-Virtual [22]	0.1036	0.2371
SchNet [51]	0.0428	0.1511
DimeNet++ [15]	0.0306	0.1214
SphereNet [37]	0.0301	0.1182
ComENet [59]	0.0326	0.1273
PaiNN [50]	0.0311	0.1208
TorchMD-Net [55]	0.0303	0.1196
GeoMFormer (ours)	**0.0252**	**0.1045**

D More Experiments

D.1 Molecule3D (Invariant)

Molecule3D [63] is a newly proposed large-scale dataset curated from the PubChemQC project [41, 44]. Each molecule has the DFT-calculated equilibrium geometric structure. The task is to predict the HOMO-LUMO energy gap, which is the same as PCQM4Mv2. The dataset contains 3,899,647 molecules in total and is split into training, validation, and test sets with the splitting ratio 6 : 2 : 2. In particular, both random and scaffold splitting methods are adopted to thoroughly evaluate the in-distribution and out-of-distribution performance of geometric molecular models. Following [59], we compare our GeoMFormer with several competitive baselines. Detailed descriptions of the training settings and baselines are presented in the appendix. It can be easily seen from Table 6 that our GeoMFormer consistently outperforms all baselines on both random and scaffold split settings, e.g., 16.3% and 11.6% relative MAE reduction compared to the previous best model respectively.

We follow [59] to use several competitive baselines for comparison including GIN-Virtual [22], SchNet [51], DimeNet++ [15], SphereNet [37] which have already been introduced in previous sections. ComENet [59] proposed a message-passing layer that operates within the 1-hop neighborhood of atoms and encoded the rotation angles to fulfill global completeness. We also implement both PaiNN [50] and TorchMD-Net [55] for comparisons.

Following [59], we evaluate our GeoMFormer model on both random and scaffold splits. Our GeoMFormer model consists of 12 layers. The dimension of hidden layers and feed-forward layers is set to 768. The number of attention heads is set to 48. The number of Gaussian Basis kernels is set to 128. We use AdamW as the optimizer, and set the hyper-parameter ϵ to 1e-8 and (β_1, β_2) to (0.9,0.999). The gradient clip norm is set to 5.0. The peak learning rate is set to 3e-4. The batch size is set to 1024. The dropout ratios for the input embeddings, attention matrices, and hidden representations are set to 0.0, 0.1, and 0.1 respectively. The weight decay is set to 0.0. The model is trained for 1 million steps with a 60k-step warm-up stage. After the warm-up stage, the learning rate decays linearly to zero. The model is trained on 16 NVIDIA V100 GPUs.

D.2 MD17 (Invariant + Equivariant)

MD17 [63] consists of molecular dynamics trajectories of several small organic molecules. Each molecule has its geometric structure along with the corresponding energy and force. The task is to predict both the energy and force of the molecule's geometric structure in the current state. To evaluate the performance of models in a limited data setting, all models are trained on only 1,000 samples from which 50 are used for validation. The remaining data is used for evaluation. For each molecule, we train a separate model on data samples of this molecule only. We set the model parameter budget the same as **(author?)** [55]. Following [55], we compare our GeoMFormer with several competitive baselines: (1) SchNet [51]; (2) PhysNet [57]; (3) DimeNet [16]; (4) PaiNN [50]; (5) NequIP [3]; (6) TorchMD-Net [55]. The results are presented in Table 7. It can be easily seen that our GeoMFormer achieves competitive performance on the energy prediction task (5 best and 1 tie out of 8 molecules) and consistently outperforms the best baselines by a significantly large margin on the force prediction task, i.e., 30.6% relative force MAE reduction in average.

Table 7: Results on MD trajectories from the MD17 dataset. Scores are given by the MAE of energy predictions (kcal/mol) and forces (kcal/mol/Å). NequIP does not provide errors on energy, for PaiNN we include the results with lower force error out of training only on forces versus on forces and energy. Benzene corresponds to the dataset originally released in **(author?)** [7], which is sometimes left out from the literature. Our results are averaged over three random splits.

Molecule		SchNet	PhysNet	DimeNet	PaiNN	NequIP	TorchMD-Net	GeoMFormer
Aspirin	*energy*	0.37	0.230	0.204	0.167	-	0.123	**0.118**
	forces	1.35	0.605	0.499	0.338	0.348	0.253	**0.171**
Benzene	*energy*	0.08	-	0.078	-	-	0.058	**0.052**
	forces	0.31	-	0.187	-	0.187	0.196	**0.146**
Ethanol	*energy*	0.08	0.059	0.064	0.064	-	0.052	**0.047**
	forces	0.39	0.160	0.230	0.224	0.208	0.109	**0.062**
Malondialdehyde	*energy*	0.13	0.094	0.104	0.091	-	0.077	**0.071**
	forces	0.66	0.319	0.383	0.319	0.337	0.169	**0.133**
Naphthalene	*energy*	0.16	0.142	0.122	0.116	-	0.085	**0.081**
	forces	0.58	0.310	0.215	0.077	0.097	0.061	**0.040**
Salicylic Acid	*energy*	0.20	0.126	0.134	0.116	-	**0.093**	0.099
	forces	0.85	0.337	0.374	0.195	0.238	0.129	**0.098**
Toluene	*energy*	0.12	0.100	0.102	0.095	-	**0.074**	0.078
	forces	0.57	0.191	0.216	0.094	0.101	0.067	**0.041**
Uracil	*energy*	0.14	0.108	0.115	0.106	-	**0.095**	**0.095**
	forces	0.56	0.218	0.301	0.139	0.173	0.095	**0.068**

Table 8: Impact of the attention modules on GeoMFormer. All other hyperparameters are kept the same for a fair comparison.

Inv-Self-Attn	Inv-Cross-Attn	Equ-Self-Attn	Equ-Cross-Attn	MSE ↓
✓	✓	✓	✓	0.0047
✗	✓	✓	✓	0.0051
✓	✗	✓	✓	0.0051
✓	✓	✗	✓	0.0056
✓	✓	✓	✗	0.0054
✗	✓	✓	✗	0.0054
✓	✗	✓	✗	0.0057
✗	✓	✗	✓	0.0055
✓	✗	✗	✓	0.0057
✗	✗	✓	✗	0.0059

D.3 Ablation Studies

In this subsection, we conduct comprehensive experiments for ablation studies on each building component of our GeoMFormer model, including both self-attention and cross-attention modules ($\mathrm{Inv\text{-}Self\text{-}Attn}, \mathrm{Equ\text{-}Self\text{-}Attn}, \mathrm{Inv\text{-}Cross\text{-}Attn}, \mathrm{Equ\text{-}Cross\text{-}Attn}$), feed-forward networks ($\mathrm{Inv\text{-}FFN}, \mathrm{Equ\text{-}FFN}$), layer normalizations ($\mathrm{Inv\text{-}LN}, \mathrm{Equ\text{-}LN}$) and the structural encoding. Without loss of generality, we conduct the experiments on the N-body Simulation task.

Impact of the attention modules. As stated in Section 4, our GeoMFormer model consists of four attention modules. We conduct a series of ablation studies to evaluate their contribution to the overall performance. In particular, we consider all possible ablation configurations that involve ablating one or more of the four modules. Note that this is an equivariant prediction task, necessitating the preservation of at least one equivariant attention module. The results are presented in Table 8, which indicates that all four attention modules consistently contribute to boosting the model's performance.

Impact of the FFN. We perform ablation studies to ascertain the contribution of both invariant and equivariant FFN modules to the model's performance. Specifically, we examine all possible settings involving the ablation of one or both of the FFN modules. The results are presented in Table 9, which demonstrates that both FFN modules positively contribute to enhancing performance.

Impact of the LN. We employ invariant and equivariant LN to stabilize training. To investigate whether the invariant and equivariant LN modules improve performance, we conduct ablation studies

Table 9: Impact of the FFN modules on GeoMFormer. All other hyperparameters are kept the same for a fair comparison.

Inv-FFN	Equ-FFN	MSE ↓
✓	✓	0.0047
✗	✓	0.0049
✓	✗	0.0055
✗	✗	0.0057

Table 10: Impact of the LN modules on GeoMFormer. All other hyperparameters are kept the same for a fair comparison.

Inv-LN	Equ-LN	MSE ↓
✓	✓	0.0047
✗	✓	0.0051
✓	✗	0.0077
✗	✗	0.0073

Table 11: Impact of structural encoding on GeoMFormer. All other hyperparameters are kept the same for a fair comparison.

Structural Encoding	MSE ↓
✓	0.0047
✗	0.0072

that encompass all possible settings of ablating one or both LN modules. The results are displayed in Table 10, demonstrating that both LN modules help to enhance performance.

Impact of the Structural Encoding. We incorporate the structural encoding as a bias term when calculating attention probability in our GeoMFormer, as described in Section A. We conduct ablation studies to see if it helps boost performance. Results are shown in Table 11. It can be seen that the introduction of structural encoding leads to improved performance.

E Experimental Details

E.1 OC20 IS2RE

Baselines. We compare our GeoMFormer with several competitive baselines for learning geometric molecular representations. Crystal Graph Convolutional Neural Network (CGCNN) [60] developed novel approaches to modeling periodic crystal systems with diverse features as node embeddings. SchNet [51] leveraged the interatomic distances encoded via radial basis functions, which serve as the weights of continuous-filter convolutional layers. DimeNet++ [15] introduced the directional message passing that encodes both distance and angular information between triplets of atoms.

GemNet [14] embedded all atom pairs within a given cutoff distance based on interatomic directions, and proposed three forms of interaction to update the directional embeddings: Two-hop geometric message passing (Q-MP), one-hop geometric message passing (T-MP), and atom self-interactions. An efficient variant named GemNet-T is proposed to use cheaper forms of interaction.

SphereNet [37] used the spherical coordinate system to represent the relative location of each atom in the 3D space and proposed the spherical message passing. GNS [46] is a framework for learning mesh-based simulations using graph neural networks and can handle complex physical systems. Graphormer-3D [53] extended Graphormer[64] to learn geometric molecular representations, which encodes the interatomic distance as attention bias terms and performed well on large-scale datasets. Equiformer [35] uses the tensor product operations to build a new scalable equivariant Transformer architecture and outperforms strong baselines on the large-scale OC20 dataset [6].

Settings. As introduced in Section 5.1.1, we follow the experimental setup of Graphormer-3D [53] for a fair comparison. Our GeoMFormer model consists of 12 layers. The dimension of hidden layers and feed-forward layers is set to 768. The number of attention heads is set to 48. The number of Gaussian Basis kernels is set to 128. We use AdamW as the optimizer and set the hyper-parameter ϵ to 1e-6 and (β_1, β_2) to (0.9,0.98). The gradient clip norm is set to 5.0. The peak learning rate is set to 2e-4. The batch size is set to 128. The dropout ratios for the input embeddings, attention matrices, and hidden representations are set to 0.0, 0.1, and 0.0 respectively. The weight decay is set to 0.0. The model is trained for 1 million steps with a 60k-step warm-up stage. After the warm-up stage, the learning rate decays linearly to zero. Following **(author?)** [35], we also use the noisy node data augmentation strategy [19] to improve the performance. The model is trained on 16 NVIDIA Tesla V100 GPUs.

E.2 OC20 IS2RS

Baselines. In this experiment, we choose several competitive baselines that perform well on equivariant prediction tasks for molecules. PaiNN [50] built upon the framework of EGNN [48] to maintain both invariant and equivariant representations and further used the Hardamard product operation to transform the equivariant representations. Specialized tensor prediction blocks were also developed for different molecular properties. TorchMD-Net [55] developed an equivariant Transformer architecture by using similar Hardamard product operations and achieved strong performance on various tasks.

SpinConv [54] encoded angular information with a local reference frame defined by two atoms and used a spin convolution on the spherical representation to capture rich angular information while maintaining rotation invariance. An additional prediction head is used to perform the equivariant prediction task, GemNet-dT [14] is a variant of GemNet-T that can directly perform force prediction and other equivariant tasks, e.g., the relaxed positions in this experiment. GemNet-OC [17] is an extension of GemNet by using more efficient components and achieved better performance on OC20 tasks.

Settings. As introduced in Section 5.1.2, we adopt the direct prediction setting for comparing the ability to perform equivariant prediction tasks on OC20 IS2RS. In particular, we re-implemented the baselines and carefully trained these models for a fair comparison. Our GeoMFormer model consists of 12 layers. The dimension of hidden layers and feed-forward layers is set to 768. The number of attention heads is set to 48. The number of Gaussian Basis kernels is set to 128. We use AdamW as the optimizer and set the hyper-parameter ϵ to 1e-6 and (β_1, β_2) to (0.9,0.98). The gradient clip norm is set to 5.0. The peak learning rate is set to 2e-4. The batch size is set to 64. The dropout ratios for the input embeddings, attention matrices, and hidden representations are set to 0.0, 0.1, and 0.0 respectively. The weight decay is set to 0.0. The model is trained for 1 million steps with a 60k-step warm-up stage. After the warm-up stage, the learning rate decays linearly to zero. The model is trained on 16 NVIDIA Tesla V100 GPUs.

E.3 PCQM4Mv2

Baselines. We compare our GeoMFormer with several competitive baselines from the leaderboard of OGB Large-Scale Challenge [22]. First, we compare several message-passing neural network (MPNN) variants. Two widely used models, GCN [30] and GIN [62] are compared along with their variants with virtual node (VN) [18, 23]. Besides, we compare GINE-VN [4] and DeeperGCN-VN [34]. GINE is the multi-hop version of GIN. DeeperGCN is a 12-layer GNN model with carefully designed aggregators. The result of MLP-Fingerprint [22] is also reported. The complexity of these models is generally $\mathcal{O}(n)$, where n denotes the number of atoms.

Additionally, we compare with several Graph Transformer models, whose computational complexity is $\mathcal{O}(n^2)$. TokenGT [29] purely used node and edge representations as the input and adopted the standard Transformer architecture without graph-specific modifications. EGT [25] used global self-attention as an aggregation mechanism and utilized edge channels to capture structural information. GRPE [45] considered both node-spatial and node-edge relations and proposed a graph-specific relative positional encoding. Graphormer [64] developed graph structural encodings and integrated them into a standard Transformer model, which achieved impressive performance across several world competitions [65, 53]. GraphGPS [47] proposed a framework to integrate the positional and structural encodings, local message-passing mechanism, and global attention mechanism into the Transformer model. All these models are designed to learn 2D molecular representations.

There also exist several models capable of utilizing the 3D geometric structure information in the training set of PCQM4Mv2. Transformer-M [40] is a Transformer-based Molecular model that can take molecular data of 2D or 3D formats as input and learn molecular representations, which was widely adopted by the winners of the 2nd OGB Large-Scale Challenge. GPS++ [42] is a hybrid MPNN and Transformer model built on the GraphGPS framework [47]. It follows Transformer-M to utilize 3D atom positions and auxiliary tasks to win first place in the large-scale challenge.

Last, we include two complex models with $\mathcal{O}(n^3)$ complexity. GEM-2 [36] used multiple branches to encode the full-range interactions between many-body objects and designed an axial attention mechanism to efficiently approximate the interaction with low computational cost. Uni-Mol+ [39] proposed an iterative prediction framework to achieve accurate quantum property prediction. It first generated 3D geometric structures from the 2D molecular graph using fast yet inaccurate methods, e.g., RDKit [32]. Given the inaccurate 3D structure as the input, the model is required to predict the equilibrium structure in an iterative manner. The predicted equilibrium structure is used to predict the quantum property. Uni-Mol+ simultaneously maintain both atom representations and pair representations, which induce the triplet complexity when updating the pair representations. With the carefully designed training strategy, Uni-Mol+ achieves state-of-the-art performance on PCQM4Mv2 while yielding high computational costs.

Settings. As previously stated, DFT-calculated equilibrium geometric structures are provided for molecules in the training set. The molecules in the validation set do not have such information. We follow Uni-Mol+ [39] to train our GeoMFormer. In particular, our model takes the RDKit-generated geometric structures as the input and is required to predict both the HOMO-LUMO energy gap and the equilibrium structure by leveraging invariant and equivariant representations respectively. After training, the model is able to predict the HOMO-LUMO gap using the RDKit-generated geometric structures. We refer the readers to Uni-Mol+ [39] for more details on the training strategies.

Our GeoMFormer model consists of 8 layers. The dimension of hidden layers and feed-forward layers is set to 512. The number of attention heads is set to 32. The number of Gaussian Basis kernels is set to 128. We use AdamW as the optimizer, and set the hyper-parameter ϵ to 1e-8 and (β_1, β_2) to (0.9,0.999). The gradient clip norm is set to 5.0. The peak learning rate is set to 2e-4. The batch size is set to 1024. The dropout ratios for the input embeddings, attention matrices, and hidden representations are set to 0.0, 0.1, and 0.1 respectively. The weight decay is set to 0.0. The model is trained for 1.5 million steps with a 150k-step warm-up stage. After the warm-up stage, the learning rate decays linearly to zero. Other hyper-parameters are kept the same as the Uni-Mol+ for a fair comparison. The model is trained on 16 NVIDIA Tesla V100 GPUs.

E.4 N-Body Simulation

Baselines. Following [48], we choose several competitive baselines for comparison. Radial Field [31] developed theoretical tools for constructing equivariant flows and can be used to perform equivariant prediction tasks. Tensor Field Network [56] embedded the position of an object in the Cartesian space into higher-order representations via products between learnable radial functions and spherical harmonics. In SE(3)-Transformer [13], the standard attention mechanism was adapted to equivariant features using operations in the Tensor Field Network model. EGNN [48] proposed a simple framework. Its invariant representations encode type information and relative distance, and are further used in vector scaling functions to transform the equivariant representations.

Settings. The input of the model includes initial positions $\mathbf{p}^0 = \{\mathbf{p}_1^0, \ldots, \mathbf{p}_5^0\} \in \mathbb{R}^{5\times3}$ of five objects, and their initial velocities $\mathbf{v}^0 = \{\mathbf{v}_1^0, \ldots, \mathbf{v}_5^0\} \in \mathbb{R}^{5\times3}$ and respective charges $\mathbf{c} = \{c_1, \ldots, c_5\} \in \{-1, 1\}^5$. We encode positions and velocities via separate equivariant streams, and updated them with separate invariant representations via cross-attention modules. The equivariant prediction is based on both equivariant representations.

We follow the settings in [48] for a fair comparison. Our GeoMFormer model consists of 4 layers. The dimension of hidden layers and feed-forward layers is set to 80. The number of attention heads is set to 8. The number of Gaussian Basis kernels is set to 64. We use Adam as the optimizer, and set the hyper-parameter ϵ to 1e-8 and (β_1, β_2) to (0.9,0.999). The learning rate is fixed to 3e-4. The batch size is set to 100. The dropout ratios for the input embeddings, attention matrices, activation functions, and hidden representations are all set to 0.4, and the drop path probability is set to 0.4. The model is trained for 10,000 epochs. The number of training samples is set to 3.000. The model is trained on 1 NVIDIA V100 GPUs.

F Broader Impacts and Limitations

This work newly proposes a general framework to learn geometric molecular representations, which has great significance in molecular modeling. Our model has demonstrated considerable positive potential for various physical and chemical applications, such as catalyst discovery and optimization, which can significantly contribute to the advancement of renewable energy processes. However, it is essential to acknowledge the potential negative impacts including the development of toxic drugs and materials. Thus, stringent measures should be implemented to mitigate these risks.

There also exist some limitations to our work. Serving as a general architecture, the ability to scale up both the model and dataset sizes is of considerable interest to the community, which has been partially explored in our extensive experiment. Additionally, our model can also be extended to encompass additional downstream invariant and equivariant tasks, which we have earmarked for future research.