From Transformers to State Spaces: GeoMamba-SE(3) for Fast and Accurate Molecular Learning

Jiayu Qin[1,2] **Zhengquan Luo**[2] **Jian Chen**[1,3] **Xuhui Li**[2] **Jiayi Chen**[4] **Zhiqiang Xu**[2]
[1]University at Buffalo [2]MBZUAI [3]Dolby Laboratories [4]Fujian Normal University

Abstract

Transformers play an important role in molecular representation learning, enabling unsupervised learning from large scale unlabeled molecule datasets. However, existing Transformer based methods suffer from heavy training computation and slow inference. To accelerate the computation and relieve the burdensome pre-training, we propose a Mamba-based framework that leverages selective state space models to learn molecular representations more efficiently. Unlike conventional methods, our model, GeoMamba-SE(3), offers streamlined computation with linear-time complexity. However, naively applying Mamba to molecules struggles with SE(3) symmetry, and representations can drift under rotations/translations—leading to chemically inconsistent features. To address this, we introduce a geometry- and statistics-aware design: (i) complete local frames at atoms by converting geometric vectors into scalar channels suitable for SSMs; (ii) multi-stream Mamba blocks are modulated by SE(3)-invariant scalars to preserve geometric stability; and (iii) we impose statistical symmetry constraints via orbit-kernel losses and invariant risk minimization, treating SE(3) actions and conformers as environments. This yields practical SE(3) stability without heavy high-order tensor representations. Experiments show that our method achieves new state-of-the-art performance benchmarks on the MoleculeNet datasets, while using only one-sixth of the training computation and 57% less computation for inference.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

1 INTRODUCTION

Molecular representation learning has been an important topic in machine learning and drug design community [Fang et al., 2022, Koukos et al., 2019, Liu et al., 2021, Méndez-Lucio et al., 2021, Rong et al., 2020, Wang et al., 2022]. In particular, graph neural networks (GNNs) [Scarselli et al., 2008] and transformers [Vaswani et al., 2017] have gained increasing popularity due to their great performance. By modeling molecular systems as graphs, GNNs naturally treat the set-like nature of collections of atoms, encode the interaction between atoms in node features, and update the features by passing messages between nodes.

Transformers are more popular and more powerful architecture for molecular property prediction applications, which were developed originally for language processing and rely on sequential input and output. Transformers have been a central component in the breakthrough of AlphaFold2 [Bryant et al., 2022] for protein structure prediction from the primary sequence. The expectations for extending this success to property prediction of other biomolecules has lead to a series of successful works of applying transformers to molecular representation learning, such as MAT [Maziarka et al., 2020], MolBERT [Fabian et al., 2020], K-BERT [Liu et al., 2020], and ChemBERTa-2 [Ahmad et al., 2022], RT [Born and Manica, 2023], SELFIE [Yüksel et al., 2023].

Neural networks succeed in molecular property prediction by incorporating equivariance constraints that leverage data symmetries. In 3D molecular structures, biases related to the SE(3) group — covering 3D translations and rotations — are essential. For instance, system energy remains constant with shifts, while force vectors rotate with system orientation. Physical laws are coordinate-invariant, meaning properties like energy remain unchanged under rotation, while others, like force, rotate accordingly.

For learning on molecular data, the features and learned functions should be $SE(3)$ stable with respect to geometric transformations acting on positions $\mathbf{r}$. This is because

many downstream targets are scalars (e.g., quantum and physico-chemical properties) that are invariant to rotations and translations, motivating an invariant molecular feature extractor for such tasks. Reflection equivariance is not required for molecular structures, as many molecules exhibit chirality; applying a reflection reverses chirality and yields a different structure with distinct properties. In this work, we achieve practical $SE(3)$ stability by (i) constructing complete local frames at atoms to convert geometric vectors into scalar channels, (ii) running multi-stream Mamba with selective state-space parameters modulated only by $SE(3)$-invariant scalars, and (iii) enforcing statistical symmetry via orbit-kernel regularization and invariant risk minimization, treating rotations and conformers as environments. For clarity, we use *invariance* for scalar targets (e.g., energies, gaps) and allow *equivariance* at intermediate coefficients in local frames; the literature sometimes mixes these terms.

Formally, let $\mathbf{x}$ denote a 3D molecular structure and $T \in SE(3)$ a roto-translation. For scalar targets we require: $f(T \cdot x) = f(x)$. For vector/tensor quantities, we require equivariance in local frames, i.e., $Z(T \cdot \mathbf{x}) = T \cdot Z(\mathbf{x})$ for any transformation T.

In molecular property prediction, Transformer-based models perform well, but often face prolonged training and inference times due to complex architectures and high computational demands, especially with equivariance. Models like UniMol, which takes about 23 V100 GPU days for pretraining, and Equiformer, which requires 24 A6000 GPU days for training on the OC20 dataset, need extensive resources. This high computation overhead limits real-time predictions and efficient screening, constraining their practical use despite high accuracy.

Meanwhile, a parallel line of research indicates that Mamba [Gu and Dao, 2023] has achieved a groundbreaking milestone with its linear-time inference and efficient training process by integrating the selection mechanism and hardware-aware algorithms into previous works.

Despite their widespread success in various domains, Mambas have only recently begun to be explored for molecular property prediction, and their effectiveness under symmetry constraints remains under-investigated. In this work, we demonstrate that selective state-space models can generalize well to molecular datasets when combined with explicit $SE(3)$ stability, and we present **GeoMamba-SE(3)**, a geometry- and statistics-aware variant of Mamba.

First, we construct a *multi-stream* Mamba architecture that processes both node sequences and edge sequences. We serialize molecular graphs into node and edge sequences using block-BFS neighborhoods and selected non-bonded pairs. Each atom's 3D coordinates are projected into a complete local frame, producing invariant scalar channels that feed into the Mamba encoder.

Second, to enforce $SE(3)$ stability in processing 3D molecular structures, we adopt a *statistical regularization*: orbit-kernel losses encourage invariant outputs across rotations/conformers, while invariant risk minimization treats these as multiple environments. This approach addresses both rotation and translation stability: centering molecules trivially handles translations, while orbit-sampling and kernel mean embedding enforce that rotated versions yield indistinguishable scalar readouts. Equivariance at intermediate coefficients is encouraged by aligning local-frame representations across orbits. Invariant risk minimization (IRM) further improves generalization by encouraging a single predictor to remain optimal across all such rotation/conformer environments. Our method shows strong performance on MoleculeNet benchmarks, demonstrating the effectiveness of this SE(3)-equivariant design.

We summarize our contributions as follows.

- We adapt the Mamba architecture to molecular datasets by introducing *multi-stream inputs* together with *local-frame scalarization of 3D coordinates*, yielding **GeoMamba-SE(3)** for efficient representation learning with geometry-aware inductive bias.
- We propose *statistical symmetry constraints*—orbit-kernel losses and invariant risk minimization to enforce $SE(3)$ invariance and equivariance in a distributional sense, avoiding reliance on heavy high-order irreducible representations (irreps) or brittle augmentation schemes.
- We design an efficient serialization pipeline that converts molecular graphs into node and edge sequences, augmented with local-frame invariant scalars, enabling Mamba-based linear-time modeling of both connectivity and geometry.
- Extensive experiments on MoleculeNet and QM9 show that GeoMamba-SE(3) achieves competitive or superior accuracy compared to strong equivariant baselines, while significantly reducing computation and improving stability, making it well-suited for high-throughput molecular property prediction.

These contributions highlight the potential of **GeoMamba-SE(3)** as a robust, scalable, and statistics-aware solution for molecular property prediction, bridging the gap between computational efficiency and principled handling of $SE(3)$ symmetries in 3D molecular structures.

2 RELATED WORK

Molecular Representation Learning Representation learning on large-scale unlabeled molecules has recently attracted much attention. SMILES-BERT [Wang et al., 2019] is trained on SMILES strings of molecules using BERT.

Subsequent work is mostly pretraining in 2D molecular topological graphs. MolCLR [Wang et al., 2022] applies data augmentation to molecular graphs at both node and graph levels, using a self-supervised contrastive learning strategy to learn molecular representations. In addition, several recent works try to leverage the 3D spatial information of molecules and focus on contrastive or transfer learning between the 2D topology and the 3D geometry of molecules. For example, GraphMVP [Liu et al., 2021] proposes a GNN-based contrastive learning framework between 2D topology and 3D geometry. GEM [Fang et al., 2022] uses bond angles and bond length as additional edge attributes to enhance 3D information. Uni-Mol [Zhou et al., 2023] is a universal 3D molecular pretraining framework that significantly enhances the ability to represent and expands the application scope in drug design.

SE(3) Equivariant Networks Equivariant neural networks operate on geometric tensors like type-L vectors to achieve equivariance. The central idea is to use functions of geometry built from spherical harmonics and irreducible representation features to achieve 3D rotational and translational equivariance as proposed in Tensor Field Network (TFN) [Thomas et al., 2018], which generalizes 2D counterparts [Worrall et al., 2017] to 3D Euclidean spaces. Previous works differ in the equivariant operations used in their networks. TFN [Thomas et al., 2018] and NequIP [Batzner et al., 2022] use graph convolution with linear messages, with the latter using extra equivariant gate activations [Weiler et al., 2018]. SEGNN [Brandstetter et al., 2021] introduces non-linear messages for irreducible representation features, and the non-linear messages use the same gate activation and improve linear messages. The SE (3)-Transformer [Fuchs et al., 2020] adopts an equivariant version of the dot product (DP) attention [Vaswani et al., 2017] with linear messages, and the attention can support vectors of any type L. Subsequent work on equivariant Transformers [Thölke and De Fabritiis, 2022, Le et al., 2022] follows the practice of DP attention and linear messages but uses more specialized architectures considering only type-0 and type-1 vectors.

State Space Models State-space models (SSMs) rooted in control theory have recently been adapted to deep learning for sequential data modeling, with advances such as HiPPO matrices [Gu et al., 2020] and LSSL [Gu et al., 2021b] laying the foundation for efficient sequence modeling. A major milestone was achieved with the Structured State Space Sequence Model (S4) [Gu et al., 2021a], which used linear state space equations to efficiently model long sequences. Subsequent work has focused on simplifying and optimizing the structure of S4. HTTYH [Gu et al., 2022b], DSS [Gupta et al., 2022], and S4D [Gu et al., 2022a] use a diagonal state matrix to reduce computation without sacrificing performance, while SGConv [Li et al., 2022] introduces a convolutional alternative. GSS [Mehta et al., 2022] and S5 [Smith et al., 2022] further improve S4 with parallel processing and multi-input / multi-output state-space designs, expanding the applicability of SSMs.

More recently, Mamba [Gu and Dao, 2023] revolutionized SSMs with efficient training and linear time inference, aided by selection mechanisms and hardware-aware optimizations. This has enabled innovations like MoE-Mamba [Pióro et al., 2024], which incorporates Mixture of Experts for scaling; MambaByte [Wang et al., 2024b], which excels in byte-level tasks; and Graph-Mamba [Wang et al., 2024a], enhancing graph network modeling with Mamba's efficient structure. Vision applications have also emerged, with Vision Mamba [Zhu et al., 2024] and VMamba [Liu et al., 2024] introducing modules to capture global visual context, while U-Mamba [Ma et al., 2024] shows the best performance in medical image segmentation.

Mamba in Protein and Drug Design Protein, molecular, and genomic modeling play a critical role in advancing drug discovery and biotechnology [Li et al., 2024, Scott et al., 2016]. Mamba-based models have been particularly transformative in these domains by efficiently handling long sequences [Guo and Schwaller, 2024, Peng et al., 2024, Schiff et al., 2024]. For example, PTM-Mamba [Peng et al., 2024] and ProtMamba [Sgarbossa et al., 2024] leverage Mamba's gated structure and state space models for protein language modeling, enabling efficient processing of extensive protein sequences while maintaining high accuracy. In generative molecular design, Saturn [Guo and Schwaller, 2024] uses the linear complexity of Mamba to excel in molecule simulation tasks, achieving superior efficiency in drug discovery. For genomic analysis, Caduceus [Schiff et al., 2024] extends Mamba with BiMamba and MambaDNA for bidirectional reverse complementary genomic modeling, significantly outperforming existing models.

3 METHODOLOGY

Preliminary: SE(3)-Equivariance Molecule systems are often described together with coordinates. For 3D Euclidean space, we can freely choose coordinate systems and change between them via the symmetries of 3D space: 3D translation, rotation and inversion ($\vec{r} \to -\vec{r}$). The groups of 3D translation, rotation and inversion form Euclidean group $E(3)$, with the first two forming Special Euclidean group $SE(3)$, the second being Special Orthogonal $SO(3)$, and the last two forming Orthogonal group $O(3)$. Formally, a function f mapping between vector spaces X and Y is equivariant to a group of transformation G if for any input $x \in X$, output $y \in Y$ and group element $g \in G$, we have $f\left(D_X(g)x\right) = D_Y(g)f(x)$, where $D_X(g)$ and $D_Y(g)$ are transformation matrices parametrized by g in X and Y.

Using Mamba to replace the transformer block in

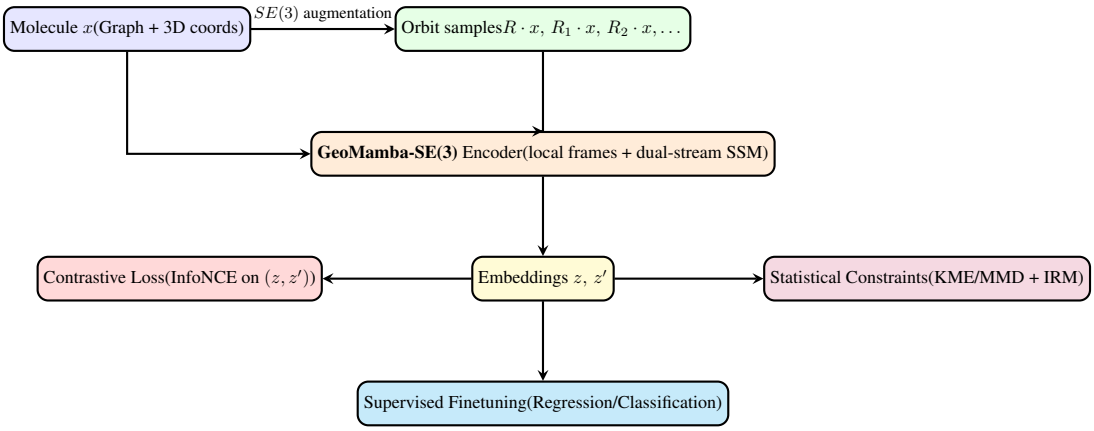

Figure 1: Overview of **GeoMamba-SE(3)**. Molecules are serialized (graph + 3D coords), orbit-rotated into multiple $SE(3)$ samples, and passed through our encoder with local-frame scalarization and dual-stream selective SSMs. Representations are trained with (i) contrastive loss for orbit-alignment, (ii) statistical $SE(3)$ constraints (KME/MMD + IRM) for stability, and (iii) supervised heads for downstream regression/classification.

Equiformer is a natural idea. However, Mamba cannot handle 3D spatial data directly as it was originally designed for NLP tasks. Although there are several recent works extending Mamba to 3D data [Liang et al., 2024], they are not specially designed for SE(3)-equivariant tasks. In addition, they are much slower than the standard Mamba due to the complex additional components.

3.1 Problem Formulation

We begin by elucidating the foundational setup and notation for molecular representation learning with **GeoMamba-SE(3)**.

Method Overview and Interplay of Components. For clarity, we describe GeoMamba-SE(3) in three stages (Fig. 1): (1) **Geometry-aware encoding:** construct $SE(3)$-equivariant local frames and scalarize 3D geometry into $SE(3)$-invariant scalar channels, then feed them into a dual-stream Mamba encoder over serialized node/edge sequences; (2) **Orbit-based representation learning:** apply random orbit augmentations (rotations and conformers) and train with an InfoNCE contrastive loss to align representations across the same molecule's orbit; (3) **Statistical symmetry constraints:** add orbit-KME/MMD and IRM penalties to statistically tighten $SE(3)$ stability beyond augmentation alone.

Terminology and Abbreviations. We spell out commonly used abbreviations at first use: BFS = breadth-first search (used for sequence serialization); KME = kernel mean embedding; MMD = maximum mean discrepancy; InfoNCE = noise-contrastive estimation objective; IRM = invariant risk minimization.

In molecular representation learning, we consider a batch of N molecules, each represented as a molecular graph $G_i = (V_i, E_i)$ with atomic features, bond features, and 3D coordinates. For each molecule, we may generate multiple *orbit samples* $g \cdot x_i$ by applying random $SE(3)$ transformations g to the coordinates. Our formulation focuses on statistical $SE(3)$ invariance: orbit samples serve as alternative "views" of the same molecule, used to impose distributional constraints. While prior work has mostly applied augmentations to 2D graphs, we leverage 3D spatial information by sampling $SE(3)$ orbit elements. These orbit-sampled molecules provide multiple stochastic views of the same structure, enabling us to enforce invariance and equivariance statistically through kernel mean embedding and IRM losses.

During orbit-based unsupervised pretraining, we treat $(x_i, g \cdot x_i)$ as positive pairs, where $g \in SE(3)$ is a random rotation or translation, and also include conformer variants of x_i as additional positives. Negatives are drawn from other molecules in the batch, or from hard negative conformers that correspond to different local minima in conformational space.This contrastive objective serves as the primary unsupervised training signal during large-scale pretraining, enabling robust representation learning without labeled supervision. Statistical symmetry constraints are incorporated as regularizers to stabilize $SE(3)$ consistency across rotations and conformers.

A neural encoder $f(x; \theta)$, instantiated as our **GeoMamba-SE(3)** backbone, processes serialized node/edge sequences with local-frame scalarization and geometry-modulated selective SSM layers. The encoder produces molecular representations z, which are optimized by the contrastive objective together with statistical $SE(3)$ regularizers.

We define similarity scores $s_{i^+} = \mathrm{sim}(z_i, z_{g \cdot i})$ for orbit-based positive pairs and $s_{ik^-} = \mathrm{sim}(z_i, z_k)$ for negatives, where $\mathrm{sim}(\cdot, \cdot)$ is cosine similarity with temperature scaling. These scores form the basis of our InfoNCE-style con-

trastive loss, which is the central unsupervised training signal, while statistical losses serve as complementary constraints. For the contrastive objective, we adopt exponential cosine similarity $\mathrm{sim}(\mathbf{z}_1, \mathbf{z}_2) \triangleq e^{\mathbf{z}_1^T \mathbf{z}_2/(\|\mathbf{z}_1\|\|\mathbf{z}_2\|\tau)}$, where τ is a temperature parameter.

Input Formulation Molecules can be represented as SMILES strings, SELFIES, or molecular graphs. While SMILES and SELFIES provide sequential encodings, molecular graphs more directly reflect the chemical structure, with atoms as nodes and bonds as edges. In this work, we adopt a hybrid representation that combines graph topology with 3D atomic coordinates. This design leverages the discrete chemical connectivity while also encoding geometric information from conformers, ensuring the model has access to both chemical and spatial inductive biases.

Following standard practice, we generate 3D conformers using ETKDG [Riniker and Landrum, 2015] followed by MMFF optimization [Halgren, 1996] in RDKit. These coordinates are not only used as positional embeddings but also serve to construct complete local frames around each atom and to derive invariant geometric scalars (bond lengths, angles, dihedrals) that modulate the selective SSM dynamics.

Overall Architecture The GeoMamba-SE(3) backbone takes as input both node sequences and edge sequences. The model processes these parallel streams using geometry-modulated selective SSM layers. Node and edge representations interact through cross-stream fusion, enabling the encoder to capture both local chemical connectivity and long-range geometric effects.

Encode 3D Positions For each atom, we integrate chemical identity and geometric context. Atom types (element, valence, ring membership) are embedded via a lookup table into a d_{type}-dimensional vector.

Local Frames and Degenerate Neighborhoods. For geometry, we construct a complete local frame at each atom i using neighbor directions. We define the neighborhood $\mathcal{N}(i)$ using covalent bonds plus a distance cutoff $r_c = 4.5\,\text{Å}$. When $|\mathcal{N}(i)| \geq 2$, we sort neighbors by Euclidean distance and set $\mathbf{e}_1$ to be the normalized vector from atom i to its closest neighbor. We then choose the closest neighbor whose direction is not nearly colinear with $\mathbf{e}_1$ (angle at least $\theta_{\min} = 20^\circ$), orthogonalize it against $\mathbf{e}_1$ and normalize to obtain $\mathbf{e}_2$. Finally, $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$ completes an orthonormal basis. In degenerate neighborhoods where all neighbors are nearly colinear with $\mathbf{e}_1$ (angles $< \theta_{\min}$), we fall back to a PCA-based rule on the local coordinate cloud to define $\mathbf{e}_2$ and complete the frame.

We project neighbor vectors into this frame to obtain invariant scalars (bond lengths, angles, and dihedrals). These scalars modulate the SSM parameters, ensuring $SE(3)$-stability while avoiding heavy tensor representations.

Non-Bonded Pair (Edge) Selection. Beyond covalent bonds, we add "soft" non-bonded edges between atom pairs (i, j) that satisfy $\|\mathbf{x}_i - \mathbf{x}_j\| \leq r_c$ with $r_c = 4.5\,\text{Å}$. For each atom we cap the number of such non-bonded neighbors to $k_{\max} = 16$, prioritizing the closest pairs to control both model capacity and computational cost.

The atom-type embeddings and scalarized geometric scalars are concatenated and linearly projected to form the node-stream input representation. Similarly, bond types, conjugacy flags, and interatomic distances define the edge-stream input representation. Both streams are then serialized and fed into parallel Mamba blocks, with invariant-gated fusion ensuring consistent exchange of chemical and geometric information.

Mamba Block After obtaining the node-stream embeddings and edge-stream embeddings, the next step is to propagate these embeddings through multiple layers of dual-stream Mamba blocks to generate molecular representations. Here we show the formulation of one layer of our dual-stream Mamba block in Algorithm 1. We take two serialized inputs: (i) the node sequence $x_{\text{node}} \in \mathbb{R}^{B \times L_n \times d_n}$ and (ii) the edge sequence $x_{\text{edge}} \in \mathbb{R}^{B \times L_e \times d_e}$. As described in previous section, the key process in one Mamba block is to decide the following parameters: $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\Delta})$, here is our formulation:

For the hidden-state matrix $\boldsymbol{A}$ and the interval Δ, we follow the original Mamba formulation and parameterize them via HiPPO [Gu et al., 2020] and broadcasted discretization, respectively. In our dual-stream design, $\boldsymbol{A}$ is *shared* across streams in a layer, while Δ is modulated only by $SE(3)$-invariant scalars. Concretely, let the node and edge input sequences be $x_{\text{node}} \in \mathbb{R}^{B \times L_n \times d_n}$, $x_{\text{edge}} \in \mathbb{R}^{B \times L_e \times d_e}$, where B is the batch size, L_n/L_e are the node/edge sequence lengths, and d_n/d_e are feature dimensions after local-frame scalarization. We compute the selective parameters $\boldsymbol{B}$ and $\boldsymbol{C}$ separately for the node and edge streams. Given $x_{\text{node}} \in \mathbb{R}^{B \times L_n \times d_n}$ and $x_{\text{edge}} \in \mathbb{R}^{B \times L_e \times d_e}$, we apply shared position-wise projectors to each token in each stream:

$$\boldsymbol{B}_{\text{node}} = S_B^{\text{node}}(x_{\text{node}}) \in \mathbb{R}^{B \times L_n \times N},$$
$$\boldsymbol{C}_{\text{node}} = S_C^{\text{node}}(x_{\text{node}}) \in \mathbb{R}^{B \times L_n \times N}.$$

$$\boldsymbol{B}_{\text{edge}} = S_B^{\text{edge}}(x_{\text{edge}}) \in \mathbb{R}^{B \times L_e \times N},$$
$$\boldsymbol{C}_{\text{edge}} = S_C^{\text{edge}}(x_{\text{edge}}) \in \mathbb{R}^{B \times L_e \times N}.$$

Here $S_B^{\text{node}}, S_C^{\text{node}}$ and $S_B^{\text{edge}}, S_C^{\text{edge}}$ are lightweight shared MLPs (equivalently, token-wise projectors) applied independently at each sequence position. Their inputs are the node/edge stream features constructed from chemical attributes together with local-frame-derived $SE(3)$-invariant

geometric scalars, so the resulting selective parameters vary across positions while remaining stable under rigid motions.

For the step-size tensor, we let

$$\Delta = \tau_\Delta(\mathrm{Param} + S_\Delta(\mathrm{Pool}_t[\phi_{\text{node}}(t), \phi_{\text{edge}}(t)])), \quad (1)$$

where $\Delta \in \mathbb{R}^{B\times 1\times D}$, D is the state dimension, $\phi_{\text{node}}, \phi_{\text{edge}}$ denote per-position $SE(3)$-invariant scalars from the two streams, Pool_t is a permutation-invariant temporal pooling, and S_Δ is an MLP used only on invariant statistics so that Δ remains $SE(3)$-stable.

For comparison, the *original* (single-stream) Mamba computes

$$\boldsymbol{B} = S_B(x), \qquad \boldsymbol{C} = S_C(x),$$

where $\boldsymbol{B}, \boldsymbol{C} \in \mathbb{R}^{B\times L\times N}$. In contrast, our formulation keeps the same backbone but replaces x by the two serialized streams and assigns *separate* projectors $S^{\text{node}}_{(\cdot)}$ and $S^{\text{edge}}_{(\cdot)}$, while sharing $\boldsymbol{A}$ and modulating Δ using only invariant scalars, thus preserving $SE(3)$ stability without changing Mamba's linear-time complexity.

Algorithm 1 One layer of GeoMamba-SE(3) dual-stream block

Get a batch of molecules; build node sequence x_{node} and edge sequence x_{edge} from features and local-frame scalars
$x_{\text{node}} : (B, L_n, d_n), \quad x_{\text{edge}} : (B, L_e, d_e)$
Hidden state $\boldsymbol{A} : (D, N) \leftarrow$ Parameter (shared across streams)
$\boldsymbol{B}_{\text{node}} \leftarrow S^{\text{node}}_B(x_{\text{node}}), \boldsymbol{C}_{\text{node}} \leftarrow S^{\text{node}}_C(x_{\text{node}})$
$\boldsymbol{B}_{\text{edge}} \leftarrow S^{\text{edge}}_B(x_{\text{edge}}), \boldsymbol{C}_{\text{edge}} \leftarrow S^{\text{edge}}_C(x_{\text{edge}})$
Interval $\Delta \leftarrow \tau_\Delta(\text{Parameter} + S_\Delta(x_{\text{node}}, x_{\text{edge}}))$
$\overline{A}, \overline{B}_{\text{node}} \leftarrow \mathrm{discretize}(\Delta, A, B_{\text{node}})$
$\overline{A}, \overline{B}_{\text{edge}} \leftarrow \mathrm{discretize}(\Delta, A, B_{\text{edge}})$
$y_{\text{node}} \leftarrow \mathrm{SSM}(\overline{A}, \overline{B}_{\text{node}}, C_{\text{node}})(x_{\text{node}})$
$y_{\text{edge}} \leftarrow \mathrm{SSM}(\overline{A}, \overline{B}_{\text{edge}}, C_{\text{edge}})(x_{\text{edge}})$
$y \leftarrow \mathrm{CrossStreamFusion}(y_{\text{node}}, y_{\text{edge}})$
Return y

3.2 Equivariant Contrastive Learning

For a molecule x, we sample a random transformation $R \in SE(3)$ and construct an augmented view $x' = R \cdot x$. We treat (x, x') as a positive pair and views from different molecules as negatives. Let z and z' denote the corresponding representations. We then optimize the standard InfoNCE loss with cosine similarity and temperature τ:

$$\mathrm{sim}(z, z') = \exp\left(\frac{\langle z, z'\rangle}{\|z\|\,\|z'\|\,\tau}\right).$$

Each molecule $\mathcal{M}$ and its augmented version $\mathcal{M}'$ form a positive pair, while views from different molecules serve as negatives. Let $f(\mathcal{M})$ and $f(\mathcal{M}')$ denote the corresponding embeddings. We use the standard InfoNCE objective:

$$s_{ij} = \mathrm{sim}(f(\mathcal{M}_i), f(\mathcal{M}'_j)),$$
$$\mathcal{L}_{\text{contrastive}} = -\sum_{i=1}^{N} \log \frac{\exp(s_{ii}/\tau)}{\sum_{j=1}^{N} \exp(s_{ij}/\tau)}.$$

Here, N is the number of molecules in the batch, $\mathrm{sim}(\cdot,\cdot)$ denotes the similarity function, and τ is the temperature parameter.

3.3 Statistical Constraints for $SE(3)$ Stability

While the orbit-contrastive loss encourages invariance by aligning embeddings of rotated molecules, it does not provide a formal guarantee that the learned representations are stable under all $SE(3)$ transformations. To address this, we introduce *statistical constraints* that regularize the representation distribution across orbits using kernel methods and invariant risk minimization.

Orbit-Induced Distributions. For a molecule $x = (G, \{\mathbf{p}_i\})$ and a rotation $R \in SO(3)$, let $f_\theta(x)$ denote the representation produced by the encoder. The *orbit distribution* is defined as

$$P_x = \{f_\theta(R \cdot x) \;:\; R \sim \nu\},$$

where ν is a distribution approximating the Haar measure over $SO(3)$. Ideally, P_x should collapse to a single point for invariant outputs, or transform consistently for equivariant outputs. We enforce this statistically using kernel mean embeddings and IRM.

KME/MMD Invariance Loss. We embed the orbit distribution into a reproducing kernel Hilbert space (RKHS) with kernel $k(\cdot,\cdot)$. Its mean embedding is

$$\mu_{P_x} = \mathbb{E}_{R\sim\nu}[\,k(f_\theta(R \cdot x), \cdot)\,].$$

For scalar predictions $r_\theta(x)$, we penalize dispersion across orbit samples via maximum mean discrepancy (MMD). We use a Gaussian RBF kernel $k(a, b) = \exp\big(-\|a - b\|^2/(2\sigma^2)\big)$, where the bandwidth σ is chosen by the median heuristic on a held-out subset of orbit samples and then fixed across runs. We draw $m = 64$ orbit samples per molecule and set $\lambda_{\text{inv}} = 0.1$ in all main experiments.

Concretely, the MMD penalty is:

$$\mathcal{L}_{\text{inv}}(x) = \frac{1}{m^2}\sum_{a,b=1}^{m} k\big(r_\theta(R_a \cdot x), r_\theta(R_b \cdot x)\big)$$
$$-\frac{2}{m}\sum_{a=1}^{m} k\big(r_\theta(R_a \cdot x), \bar{r}_\theta(x)\big),$$

where $\bar{r}_\theta(x) = \mathbb{E}_R[r_\theta(R \cdot x)]$. With a characteristic kernel (e.g., Gaussian), this loss forces orbit distributions to collapse, ensuring invariance in expectation.

Invariant Risk Minimization (IRM). We treat each rotated or conformational variant of a molecule as a separate *environment* $e \in \mathcal{E}$. Let $\Phi(x)$ be the learned representation and w a linear predictor. The IRM penalty is

$$\min_{\Phi, w} \sum_{e \in \mathcal{E}} R_e(w \circ \Phi) \; + \; \lambda_{\text{irm}} \sum_{e \in \mathcal{E}} \left\| \nabla_w R_e(w \circ \Phi) \big|_{w=1.0} \right\|^2,$$

where R_e is the risk within environment e. This encourages Φ to admit a single predictor w that is simultaneously optimal across all environments, thus improving generalization stability.

Here we adopt the IRM relaxation where the linear head w is reparameterized and *fixed* at $w = 1.0$ during the penalty computation; this removes trivial rescalings and penalizes only directionally suboptimal heads across environments, following standard practice in IRM.

Total objective. Combining the contrastive pretraining loss with statistical constraints yields the full objective:

$$\mathcal{L} = \mathcal{L}_{\text{contrastive}} + \lambda_{\text{inv}} \, \mathbb{E}_x[\mathcal{L}_{\text{inv}}(x)] + \lambda_{\text{irm}} \, \text{IRM}.$$

This ensures that representations are learned primarily through orbit-contrastive alignment, while being statistically stabilized under the full $SE(3)$ group via KME/MMD and IRM penalties.

Physical–statistical intuition. For a fixed molecule x, random roto-translations $R \in \text{SO}(3)$ generate an orbit $\{R \cdot x\}$ that should map to indistinguishable scalar outputs for invariant targets. Our use of kernel mean embeddings (KME) with a *characteristic* kernel implies that two distributions coincide if and only if their RKHS mean embeddings coincide. Hence, minimizing an orbit-MMD between $\{f_\theta(R \cdot x)\}$ and its orbit-average $\bar{f}_\theta(x)$ statistically *collapses* the orbit-induced law to a point mass, guaranteeing invariance in expectation under mild sampling noise. This turns SE(3) stability into a verifiable distributional property rather than a brittle architectural constraint.

4 EXPERIMENTS AND RESULTS

4.1 Comparison of Efficiency

Hardware-Matched Throughput. To avoid mixed-hardware comparisons, we re-measure throughput under matched settings on the **same NVIDIA A6000 GPU** with identical data preprocessing, batch sizes, and pretraining corpus. Under these matched settings, we obtain:

- Uni-Mol (base; 384-dim, 6 layers, 12 heads): 31.73 it/s
- GeoMamba (384-dim, 10 layers): 67.55 it/s
- Uni-Mol (large; 768-dim, 12 layers, 24 heads): 6.73 it/s
- GeoMamba (512-dim, 10 layers): 59.64 it/s

Since selective state space models (SSMs) are relatively new in molecular property learning, we compare GeoMamba-SE(3) not only with strong Transformer / GNN baselines (e.g. Uni-Mol, GEM, HiMol) but also with existing SSM-based methods (e.g. GraphMamba, MOL-Mamba [Hu et al., 2024]). We conduct evaluation under a two-stage regime: (i) orbit-contrastive pretraining on large unlabeled molecular corpora, then (ii) supervised fine-tuning on downstream property prediction tasks (classification/regression).We thus benchmark GeoMamba-SE(3) against both established baselines and SSM-derived methods under the same data splits and protocols on MoleculeNet and QM9.

4.2 Experiments on MoleculeNet Dataset

MoleculeNet [Wu et al., 2018] is a popular benchmark for molecular property prediction, including data sets focusing on different molecular properties, from quantum mechanics and physical chemistry to biophysics and physiology.

Reporting Protocol. Unless otherwise stated, we report mean $\pm$ standard deviation over **3 random seeds** for GeoMamba-SE(3) on the main benchmarks (same data split). Baseline numbers are taken from prior work when multi-seed statistics are unavailable.

We follow Uni-Mol's $\sim$19M corpus. SMILES canonicalization and salt-stripping are applied; duplicates are removed by InChIKey. For each molecule we attempt up to n_{conf}=10 ETKDG conformer trials with MMFF relaxation; failures fall back to n_{conf}=1 or are skipped (skip rate $< 1.2\%$). Coordinates are standardized (centered) before local-frame construction. We retain the same training/validation/test splits as Uni-Mol.

For MoleculeNet pretraining and fine-tuning, we replace Transformer encoders with our GeoMamba-SE(3) architecture. During contrastive pretraining, we derive the molecule-level embedding via pooled fusion of node and edge final representations. Our encoder uses 12 dual-stream Mamba layers, each with hidden dimension 512. The node-stream and edge-stream embedding dimensions are set to 256 and 256 scalar channels after local-frame scalarization. For downstream fine-tuning, we tune the weights of statistical regularizers ($\lambda_{inv}, \lambda_{eq}, \lambda_{irm}$). We adopt ROC-AUC for classification tasks and RMSE/MAE for regression tasks. All experiments are run on NVIDIA A6000 GPUs.

We report the results on 7 classification baselines and 5 regression baselines, as done in MolCLR and GraphMVP. As shown in Tables 1 and 2, compared to the most convincing transformer model, our method achieves slightly better performance than Uni-Mol[Zhou et al., 2023] and GEM [Fang et al., 2022] with 7 new SOTA performance in 12 downstream tasks, also surpasses the existing contrastive learn-

Table 1: ROC-AUC on MoleculeNet classification tasks (Higher is better). For GeoMamba-SE(3), we report mean ± standard deviation over 3 random seeds (same data split). Baseline numbers are taken from prior work when multi-seed statistics are not available.

Method	Contrastive?	BBBP	BACE	ClinTox	Tox21	SIDER	HIV	MUV	Avg
D-MPNN [Yang et al., 2019]	–	71.0	80.9	90.6	75.9	57.0	77.1	78.6	75.0
AttentiveFP [Xiong et al., 2019]	–	64.3	78.4	84.7	76.1	60.6	75.7	76.6	74.3
GEM [Fang et al., 2022]	–	72.4	85.6	90.1	78.1	**67.2**	80.6	81.7	79.4
Uni-Mol [Zhou et al., 2023]	–	72.9	85.7	91.9	79.6	65.9	80.8	82.1	80.1
HiMol [Zang et al., 2024]	+	73.1	86.0	90.5	78.8	64.5	81.5	82.0	80.6
MOL-Mamba [Hu et al., 2024]	+	73.5	86.1	91.2	79.5	66.0	82.0	82.4	81.0
GraphMVP [Liu et al., 2021]	+	72.4	81.2	79.1	75.9	63.9	77.0	77.7	75.4
MolCLR [Wang et al., 2022]	+	72.2	82.4	91.2	75.0	58.9	78.1	79.6	77.7
Ours (GeoMamba-SE(3), 3 seeds)	+	**74.2** ± 0.7	**87.3** ± 1.1	**92.4** ± 0.7	**80.3** ± 0.4	65.8 ± 0.1	**83.0** ± 0.4	**83.5** ± 0.3	**82.1** ± 0.5

Table 2: Regression performance on MoleculeNet (Lower is better). For GeoMamba-SE(3), we report mean ± standard deviation over 3 random seeds (same data split). Baseline numbers are taken from prior work when multi-seed statistics are not available.

Method	ESOL	FreeSolv	Lipo	QM7	QM8	Avg (RMSE/MAE)
D-MPNN [Yang et al., 2019]	1.050	2.082	0.683	103.5	0.0190	1.272
GROVERlarge [Rong et al., 2020]	0.895	2.272	0.823	92.0	0.0224	1.330
GEM [Fang et al., 2022]	0.798	1.877	0.660	58.9	0.0171	1.112
Uni-Mol [Zhou et al., 2023]	0.788	1.480	0.603	**41.8**	0.0156	0.957
HiMol [Zang et al., 2024]	0.770	1.400	0.620	45.0	0.0152	0.965
MOL-Mamba [Hu et al., 2024]	0.745	1.450	0.610	42.5	0.0150	0.905
MolCLR [Wang et al., 2022]	1.271	2.594	0.691	66.8	0.0178	1.519
GraphMVP [Liu et al., 2021]	1.029	–	0.681	–	–	–
Ours (GeoMamba-SE(3), 3 seeds)	**0.702** ± 0.009	**1.327** ± 0.143	**0.571** ± 0.019	59.9 ± 1.4	**0.0149** ± 0.0003	**0.844**

Table 3: QM9 performance on HOMO–LUMO gap (MAE in eV, lower is better). For GeoMamba-SE(3), we report mean ± standard deviation over 3 seeds.

Method	MAE
D-MPNN	0.00814
GEM	0.00746
Uni-Mol	0.00685
HiMol	0.00680
MOL-Mamba	0.00678
Ours (GeoMamba-SE(3), 3 seeds)	**0.00673** ± 0.00018

Table 4: Training speed and memory usage comparison

	Time Cost(GPU days)	Memory Usage (GB)
Uni-mol	23	23.4
Ours	4	9.8

Table 5: Inference speed and memory usage comparison

	Speed(molecule per second)	Max batch size
Uni-mol	126	378
Ours	209	1072

Table 6: Ablations on MoleculeNet (classification), mean ROC-AUC↑ (±std over 3 seeds).

Variant	ROC-AUC (%)
Full GeoMamba-SE(3)	**82.1** ± 0.2
w/o Statistical Constraints	79.4 ± 0.3
w/o IRM (only KME/MMD)	80.7 ± 0.2
w/o KME/MMD (only IRM)	80.2 ± 0.2
Single-stream (node only)	79.9 ± 0.3

ing models, GraphMVP and MolCLR, by a large margin, showcasing the effectiveness of our method.

In addition to Uni-Mol and GEM, we further compare against HiMol [Zang et al., 2024], MOL-Mamba [Hu et al., 2024], and GraphMVP / MolCLR, to show how our method fares against both architecture-based and SSM-based competitors.

4.3 Experiments on QM9 dataset

QM9 is a widely used benchmark for the prediction of quantum chemical properties, which includes 12 distinct tasks. For simplicity and consistency with previous work, we focus our experiments on a single downstream task: predicting the energy gap between the HOMO and LUMO orbitals of a molecule, a key property in quantum chemistry. As shown in Table 3, our proposed method achieves superior performance compared to existing state-of-the-art methods, highlighting the effectiveness of our approach.

In the fine-tuning stage, we use the mean squared error (MSE) as the loss function to optimize the predictions. To ensure a fair and comprehensive evaluation, we perform a thorough hyperparameter search, detailed in Table 9. Im-

Table 7: Ablations: efficiency metrics at batch size 64.

Variant	Params (M)	FLOPs (G)	Mem (GB)
Full GeoMamba-SE(3)	52	38.4	9.8
w/o Edge stream	41	31.2	7.6
w/o Statistical constraints†	52	34.5	8.9

† excludes extra forward passes for orbit samples (m=0).

Table 8: OGB-PCQM4Mv2 performance (MAE, ↓). Mean ± std over 3 seeds.

Method	Val MAE (↓)	Test-dev MAE (↓)
GeoMamba (Ours)	**0.0656** ± 0.0044	0.0702 ± 0.0014
EGT + Tri. Attn.	0.0686	**0.0698**
GraphGPT-L48	0.0682	0.0709

portantly, the same hyperparameter search space is applied across both MoleculeNet and QM9 experiments, ensuring consistency in our methodology.

4.4 Explicit Pointwise $SE(3)$ Invariance Diagnostics

Beyond task-level performance, we directly probe representation-level $SE(3)$ stability for the pretrained GeoMamba-SE(3) encoder by comparing embeddings of original and randomly rotated molecules. We detail the setup and metrics in Appendix B.1. Empirically, we observe cosine similarity $\approx$ 1.0000 and mean squared ℓ_2 difference ≈ 0.0000 (within numerical precision), indicating that learned embeddings are effectively identical under random rigid motions.

4.5 Large-Scale Benchmark: OGB-PCQM4Mv2

To assess scalability on a large graph-level molecular benchmark, we evaluate GeoMamba-SE(3) on OGB-PCQM4Mv2 following the standard split and report MAE on the validation and test-dev sets. Results are averaged over 3 seeds.

Here we discuss the efficiency of our model, here we consider both the efficiency in training and inference time. For training efficiency, we measure throughput with a batch size of 64 under identical data preprocessing. For inference, we focus on the encoder layers (Transformer or Mamba backbone), excluding dataset I/O overhead. As shown in Tables 4 and 5, GeoMamba-SE(3) achieves ∼6× faster training speed and 57% lower memory usage in pretraining. At inference time, the method improves throughput by ∼40% and allows 65% larger maximum batch size.

Beyond throughput and memory, GeoMamba-SE(3) also reduces multiple counts: our encoder has P= 52M parameters vs. 61M for the Transformer backbone and ∼38% fewer FLOPs per forward under the same tokenization. We further stratify inference throughput by molecule size (atoms per graph) and observe near-constant per-token throughput across bins $[10, 20), [20, 40), [40, 80)$.

Table 9: Search space for MoleculeNet and QM9 experiments

Hyperparameter	Space for searchings
Learning rate	$[2e-5, 1e-4]$
Batch size	$[128, 256]$
Epochs	$[20, 40]$
Dropout	$[0.0, 0.1]$
Warmup ratio	$[0.0, 0.006]$

4.6 Ablation Studies

To better understand the contribution of each design choice in GeoMamba-SE(3), we perform controlled ablation studies on MoleculeNet (classification and regression) and QM9 (HOMO–LUMO). We examine the effect of (i) removing statistical symmetry constraints, (ii) removing IRM regularization, (iii) disabling the edge stream.

Takeaways. Removing KME/MMD+IRM harms accuracy most (Table 6), confirming their role in SE(3) stability. Dropping the edge stream saves ∼21% FLOPs and memory (Table 7) but loses 2.2 ROC-AUC, revealing a clear accuracy–efficiency trade-off.

The results highlight three key observations. First, removing statistical constraints significantly degrades performance, confirming that KME/MMD and IRM are crucial for SE(3) stability. Second, IRM provides an additional improvement over KME/MMD alone, particularly in low-data regimes. Third, dual-stream design (node + edge) outperforms single-stream by +2.2 ROC-AUC, validating the importance of edge-level information.

5 CONCLUSION

We introduced **GeoMamba-SE(3)**, a Mamba-based architecture with local-frame scalarization, dual-stream selective SSMs, and statistical symmetry constraints (KME/MMD + IRM) for molecular property prediction. GeoMamba-SE(3) attains practical SE(3) stability without heavy high-order tensor representations and achieves competitive or superior accuracy with improved efficiency on MoleculeNet and QM9. Ablations confirm the importance of statistical constraints and the edge stream. Future work will add lightweight tensor heads for vector/tensor targets and extend pretraining/transfer to larger biomolecules and tasks.

References

Waqar Ahmad, Emmanuel Simon, Sam Chithrananda, et al. Chemberta-2: Towards chemical foundation models. *arXiv preprint arXiv:2209.01712*, 2022.

Simon Batzner, Albert Musaelian, Linfeng Sun, et al. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):2453, 2022.

Jannis Born and Matteo Manica. Regression transformer enables concurrent sequence regression and generation for molecular language modelling. *Nature Machine Intelligence*, 5(4):432–444, 2023.

Johannes Brandstetter, Rianne Hesselink, Els van der Pol, et al. Geometric and physical quantities improve e (3) equivariant message passing. *arXiv preprint arXiv:2110.02905*, 2021.

Patrick Bryant, Gabriele Pozzati, and Arne Elofsson. Improved prediction of protein-protein interactions using alphafold2. *Nature Communications*, 13(1):1265, 2022.

Benedek Fabian et al. Molecular representation learning with language models and domain-relevant auxiliary tasks. *arXiv preprint arXiv:2011.13230*, 2020.

Xiang Fang, Liang Liu, Jian Lei, et al. Geometry-enhanced molecular representation learning for property prediction. *Nature Machine Intelligence*, 4 (2):127–134, 2022.

Fabian Fuchs, Daniel Worrall, Volker Fischer, et al. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in neural information processing systems*, 33:1970–1981, 2020.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Albert Gu, Tri Dao, Stefano Ermon, et al. Hippo: Recurrent memory with optimal polynomial projections. *Advances in neural information processing systems*, 33:1474–1487, 2020.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021a.

Albert Gu, Isys Johnson, Karan Goel, et al. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021b.

Albert Gu, Karan Goel, Ankit Gupta, et al. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022a.

Albert Gu, Isys Johnson, Abhishek Timalsina, et al. How to train your hippo: State space models with generalized orthogonal basis projections. *arXiv preprint arXiv:2206.12037*, 2022b.

Jeff Guo and Philippe Schwaller. Saturn: Sample-efficient generative molecular design using memory manipulation. *arXiv preprint arXiv:2405.17066*, 2024.

Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. *Advances in Neural Information Processing Systems*, 35:22982–22994, 2022.

Thomas A. Halgren. Merck molecular force field. i. basis, form, scope, parameterization, and performance of mmff94. *Journal of Computational Chemistry*, 17, 1996. URL https://api.semanticscholar.org/CorpusID:7378729.

Jingjing Hu, Dan Guo, Zhan Si, Deguang Liu, Yunfeng Diao, Jing Zhang, Jinxing Zhou, and Meng Wang. Mol-mamba: Enhancing molecular representation with structural & electronic insights. 2024. arXiv preprint arXiv:2412.16483.

Panagiotis I Koukos, L C Xue, and Alexandre MJJ Bonvin. Protein–ligand pose and affinity prediction: Lessons from d3r grand challenge 3. *Journal of Computer-Aided Molecular Design*, 33:83–91, 2019.

Thao Le, Frank Noé, and Djork-Arné Clevert. Equivariant graph attention networks for molecular property prediction. *arXiv preprint arXiv:2202.09891*, 2022.

Jiatong Li, Yunqing Liu, Wenqi Fan, Xiao-Yong Wei, Hui Liu, Jiliang Tang, and Qing Li. Empowering molecule discovery for molecule-caption translation with large language models: A chatgpt perspective. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

Yutong Li, Tianle Cai, Yilun Zhang, et al. What makes convolutional models great on long sequence modeling? *arXiv preprint arXiv:2210.09298*, 2022.

Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. In *Advances in Neural Information Processing Systems*, 2024.

Shengchao Liu, Huixuan Wang, Wen Liu, et al. Pre-training molecular graph representation with 3d geometry. *arXiv preprint arXiv:2110.07728*, 2021.

Weijie Liu, Peng Zhou, Zhe Zhao, et al. K-bert: Enabling language representation with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2901–2908, 2020.

Yang Liu, Yanchao Tian, Yue Zhao, et al. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024.

Jiawei Ma, Fang Li, and Bing Wang. U-mamba: Enhancing long-range dependency for biomedical

image segmentation. *arXiv preprint arXiv:2401.04722*, 2024.

Lukasz Maziarka, Tomasz Danel, Szymon Mucha, et al. Molecule attention transformer. *arXiv preprint arXiv:2002.08264*, 2020.

Harsh Mehta, Ankit Gupta, Ashwin Cutkosky, et al. Long range language modeling via gated state spaces. *arXiv preprint arXiv:2206.13947*, 2022.

Omar Méndez-Lucio, Mohammad Ahmad, Emilio A del Rio-Chanona, et al. A geometric deep learning approach to predict binding conformations of bioactive molecules. *Nature Machine Intelligence*, 3(12): 1033–1039, 2021.

Zhangzhi Peng, Benjamin Schussheim, and Pranam Chatterjee. Ptm-mamba: A ptm-aware protein language model with bidirectional gated mamba blocks. *bioRxiv*, pages 2024–02, 2024.

Mateusz Pióro, Kamil Ciebiera, Konrad Król, et al. Moe-mamba: Efficient selective state space models with mixture of experts. *arXiv preprint arXiv:2401.04081*, 2024.

Sereina Riniker and Gregory A. Landrum. Better informed distance geometry: Using what we know to improve conformation generation. *Journal of Chemical Information and Modeling*, 55(12):2562–2574, 2015. doi: 10.1021/acs.jcim.5b00654. PMID: 26575315.

Yu Rong, Yatao Bian, Tingyang Xu, et al. Self-supervised graph transformer on large-scale molecular data. *Advances in Neural Information Processing Systems*, 33:12559–12571, 2020.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, et al. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

Yair Schiff, Chia-Hsiang Kao, Aaron Gokaslan, Tri Dao, Albert Gu, and Volodymyr Kuleshov. Caduceus: Bi-directional equivariant long-range dna sequence modeling. *arXiv preprint arXiv:2403.03234*, 2024.

Duncan E Scott, Andrew R Bayly, Chris Abell, and John Skidmore. Small molecules, big targets: drug discovery faces the protein–protein interaction challenge. *Nature Reviews Drug Discovery*, 15(8):533–550, 2016.

Damiano Sgarbossa, Cyril Malbranke, and Anne-Florence Bitbol. Protmamba: A homology-aware but alignment-free protein state space model. *bioRxiv*, pages 2024–05, 2024.

J T H Smith, Alex Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022.

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

Nathaniel Thomas, Tess Smidt, Steven Kearnes, et al. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Chen Wang, Oleksii Tsepa, Jiawei Ma, et al. Graph-mamba: Towards long-range graph sequence modeling with selective state spaces. *arXiv preprint arXiv:2402.00789*, 2024a.

Jinghan Wang, Tharun Gangavarapu, Jin Ning Yan, et al. Mambabyte: Token-free selective state space model. *arXiv preprint arXiv:2401.13660*, 2024b.

Shuangjia Wang, Yibo Guo, Yuyang Wang, et al. Smiles-bert: large scale unsupervised pre-training for molecular property prediction. In *Proceedings of the 10th ACM international conference on bioinformatics, computational biology and health informatics*, pages 429–436, 2019.

Yifei Wang, Jie Wang, Zhen Cao, et al. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence*, 4(3):279–287, 2022.

Maurice Weiler, Mario Geiger, Max Welling, et al. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems*, 31, 2018.

Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, et al. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5028–5037, 2017.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, et al. Moleculenet: a benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018.

Zhaocheng Xiong, De Wang, Xia Liu, et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *Journal of Medicinal Chemistry*, 63(16):8749–8760, 2019.

Kevin Yang, Kyle Swanson, Wengong Jin, et al. Analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling*, 59(8):3370–3388, 2019.

Alperen Yüksel, Ece Ulusoy, Ali Ünlü, et al. Selformer: molecular representation learning via selfies language models. *Machine Learning: Science and Technology*, 4 (2):025035, 2023.

Xuan Zang, Xianbing Zhao, and Buzhou Tang. Hierarchical molecular graph self-supervised learning for property prediction. 2024.

Guangxu Zhou, Zilong Gao, Qi Ding, et al. Uni-mol: A universal 3d molecular representation learning framework. 2023.

Lin Zhu, Bing Liao, Qiao Zhang, et al. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024.

A Reproducibility Checklist

1. This paper:
   - Includes a conceptual outline and/or pseudocode description of AI methods introduced (**yes**/partial/no/NA)
   - Clearly delineates statements that are opinions, hypotheses, and speculation from objective facts and results (**yes**/no)
   - Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (**yes**/no)
2. Does this paper make theoretical contributions? (**yes**/no)
   If yes, please complete the list below:
   - All assumptions and restrictions are stated clearly and formally. (**yes**/partial/no)
   - All novel claims are stated formally (e.g., in theorem statements). (**yes**/partial/no)
   - Proofs of all novel claims are included. (**yes**/partial/no)
   - Proof sketches or intuitions are given for complex and/or novel results. (**yes**/partial/no)
   - Appropriate citations to theoretical tools used are given. (**yes**/partial/no)
   - All theoretical claims are demonstrated empirically to hold. (**yes**/partial/no/NA)
   - All experimental code used to eliminate or disprove claims is included. (**yes**/no/NA)
3. Does this paper rely on one or more datasets? (**yes**/no)
   If yes, please complete the list below:
   - A motivation is given for why the experiments are conducted on the selected datasets. (**yes**/partial/no/NA)
   - All novel datasets introduced in this paper are included in a data appendix. (**yes**/partial/no/NA)
   - All novel datasets introduced in this paper will be made publicly available upon publication with a license allowing free research use. (yes/partial/no/**NA**)
   - All datasets drawn from the existing literature are accompanied by appropriate citations. (**yes**/no/NA)
   - All datasets drawn from the existing literature are publicly available. (**yes**/partial/no/NA)
   - Datasets that are not publicly available are described in detail, with justification. (yes/partial/no/**NA**)
4. Does this paper include computational experiments? (**yes**/no)
   If yes, please complete the list below:

- Number/range of values tried per (hyper-)parameter and selection criteria are reported. (**yes**/partial/no/NA)
- Code for data preprocessing is included in the appendix. (yes/partial/**No**)
- Source code for conducting and analyzing experiments is included. (**yes**/partial/no)
- Code will be released publicly upon publication with a permissive license. (**yes**/partial/no)
- Code includes comments with implementation details and paper references. (**yes**/partial/no)
- Seed setting methods for stochastic algorithms are described. (**yes**/partial/no/NA)
- Computing infrastructure (hardware/software specs) is reported. (**yes**/partial/no)
- Evaluation metrics are formally described with motivations. (**yes**/partial/no)
- Number of runs per result is specified. (**yes**/no)
- Performance analysis includes variation, confidence, or distributions. (**yes**/no)
- Significance of performance differences is assessed with statistical tests. (**yes**/partial/no)
- Final (hyper-)parameter settings are listed. (**yes**/partial/no/NA)

B Additional Camera-Ready Clarifications

B.1 Pointwise $SE(3)$ Invariance Diagnostics

To complement task-level evaluations, we directly probe $SE(3)$ stability at the representation level for the *pre-trained* GeoMamba-SE(3) encoder.

Setup. We randomly sample a batch of molecules from the test split. For each molecule, we draw a random 3D rotation $R \in SO(3)$ (e.g., by sampling a random axis and angle) and obtain a rotated copy of the coordinates. We then run the encoder once on the original molecule and once on the rotated molecule to obtain embeddings z and z_R.

Metrics. We report (i) cosine similarity $\cos(z, z_R)$ and (ii) mean squared ℓ_2 difference $\|z - z_R\|_2^2$. Ideal invariance corresponds to cosine $\to 1$ and $\ell_2 \to 0$.

Protocol and Reproducibility. We keep preprocessing and batching identical between the original and rotated molecules, and we fix the random seed used for sampling rotations. We will release a script and a one-command reproduction recipe together with the code release.

B.2 Scope of Statistical Symmetry Constraints (KME/MMD + IRM)

Our statistical constraints are designed to *tighten* $SE(3)$ invariance in a distributional sense rather than to claim strict, layer-wise tensor equivariance. We therefore avoid overclaiming "formal guarantees" in the sense of exact equivariant operators. Instead, the orbit-MMD/KME term explicitly penalizes dispersion of orbit-induced scalar predictions, and the IRM penalty encourages a single predictor to remain optimal across rotation/conformer environments, improving robustness and generalization.

C Limitations

Experiment Our experiments cover MoleculeNet, QM9, and a large-scale graph benchmark (OGB-PCQM4Mv2), but they may still not fully represent the variety of molecular structures and tasks in real-world applications. Additionally, we did not test on very large, complex molecules with extremely long token counts, meaning we have yet to fully capitalize on GeoMamba-SE(3)'s linear complexity for handling much longer sequences. Future work should expand testing to more diverse datasets and larger molecules to better assess scalability and generalizability.

Methodology Our approach emphasizes statistical invariance via orbit-based KME/MMD and IRM penalties, rather than enforcing full tensor-level equivariance. This design streamlines computational demands for downstream tasks. However, fully equivariant architectures may still benefit applications where orientation matters, such as molecular–protein binding pose estimation.

Application Restricting our GeoMamba-SE(3) model to molecular classification and regression tasks may not fully utilize its potential. Unlike models in computer vision and NLP that display strong zero-shot capabilities, our model's ability to generalize to unseen molecular tasks remains untested. Future work could explore zero-shot performance, demonstrating broader applicability and versatility across varied molecular challenges.