GemNet: Universal Directional Graph Neural Networks for Molecules

Anonymous Author(s)
Affiliation
Address
email

Abstract

Effectively predicting molecular interactions has the potential to accelerate molecular dynamics by multiple orders of magnitude and thus revolutionize chemical simulations. Graph neural networks (GNNs) have recently shown great successes for this task, overtaking classical methods based on fixed molecular kernels. However, they still appear very limited from a theoretical perspective, since regular GNNs cannot distinguish certain types of graphs. In this work we close this gap between theory and practice. We show that GNNs with directed edge embeddings and two-hop message passing are indeed universal approximators for predictions that are invariant to global rotation and translation, and equivariant to permutation. We then leverage these insights and multiple structural improvements to propose the geometric message passing neural network (GemNet). We demonstrate the benefits of the proposed changes in multiple ablation studies. GemNet outperforms previous models on the COLL and MD17 molecular dynamics datasets by $36\,\%$, performing especially well on the most challenging molecules.

1 Introduction

Graph neural networks (GNNs) have shown great promise for predicting the energy and other quantum mechanical properties of molecules. They can predict these properties orders of magnitudes faster than methods from quantum chemistry — at comparable accuracy. GNNs can thus enable the accurate simulation of systems that are orders of magnitude larger than with regular methods. However, they still exhibit severe theoretical and practical limitations. Regular GNNs are only as powerful as the 1-Weisfeiler Lehman test of isomorphism and thus cannot distinguish between certain molecules [42, 54]. Moreover, they require a large number of training samples to achieve good accuracy.

In this work we first resolve the questionable expressiveness of GNNs by proving sufficient conditions for universality in the case of invariance to translations and rotations and equivariance to permutations. Rotationally invariant predictions cover many important chemical tasks such as classification or energy prediction for molecular dynamics. Simply using the full geometric information (e.g. all pairwise atomic distances) in a layer does not ensure universal approximation. For example, if our model uses a rotationally invariant layer we lose the relative information between components. Such a model thus cannot distinguish between features that are rotated differently. This issue is commonly known as the "Picasso problem": An image model with rotationally invariant layers cannot detect whether a person's eyes are rotated correctly. Instead, we need a model that preserves relative rotational information and is only invariant to *global* rotations. To prove universality in this case we extend a recent universality result based on point cloud models that use representations of the rotation group $\mathrm{SO}(3)$ [16]. We prove that spherical representations are sufficient in the invariant case; full

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

$\mathrm{SO}(3)$ representations are not necessary. We then discretize spherical representations by selecting points on the sphere based on the directions to neighboring atoms. We can connect this model to GNNs by interpreting these directions as directed edge embeddings. For example, the embedding direction of atom a would be defined by atom c, resulting in the edge embedding e_{ca}. Updating the spherical representation of atom a based on atom b then corresponds to two-hop message passing between the edges e_{ca} and e_{db} via e_{ba}, with atoms c and d defining the embedding directions. This message passing formalism naturally allows us to obtain the molecule's full geometrical information (distances, angles, and dihedral angles), and the direct correspondence proves the model's universality.

We call this edge-based two-hop message passing scheme *geometric message passing*, and propose multiple structural enhancements to improve the practical performance of this formalism. Based on these changes we develop the highly accurate and sample-efficient geometric message passing neural network (GemNet). We furthermore show that stabilizing the variance of GemNet's activations with predetermined scaling factors yields significant improvements over regular normalization layers.

We investigate the proposed improvements in a range of ablation studies, and show that each of them significantly reduces the model error. These changes introduce little to no computational overhead over two-hop message passing. Altogether, GemNet outperforms previous models for force predictions on COLL by $34\,\%$ and on MD17 by $38\,\%$ on average. We observe the largest improvements for the most challenging molecules, which exhibit dynamic, non-planar geometries. In summary, our contributions are:

- Showing the **universality** of spherical representations and two-hop message passing with directed edge embeddings for rotationally invariant predictions.
- **Geometric message passing**: Symmetric message passing enhanced by geometric information.
- Incorporating all proposed improvements in the **Geometric Message Passing Neural Network (GemNet)**, which significantly outperforms previous methods for molecular dynamics prediction.

2 Related work

Machine learning potentials. Research on predicting a molecule's energy and forces (so-called machine learning potentials) started with hand-fitted analytical functions and then gradually moved towards fully learned models. Classical force fields can be viewed as the very first instances of these. They use analytical functions with coefficients that were hand-tuned based on experimental data. A popular example for these is the Merck Molecular Force Field (MMFF94) [27]. The next wave of methods uses kernel ridge regression based on fixed, hand-crafted molecular representations [3, 8, 21]. Finally, modern research mostly focusses on fully end-to-end learnable models based on GNNs [26, 49]. Note that these models can still be combined with hand-crafted molecular representations to improve performance [45]. We consider this combination as orthogonal research.

Directional GNNs. We can also achieve equivariance and invariance to rotations without relying on group representations. Directional GNNs achieve this by representing directional information explicitly [50] or in the form of angles [32] and dihedral angles [23, 36]. Our work is focused on this class of models, proving their universality and proposing an improved variant, GemNet.

Expressiveness of GNNs. A large part of GNN research has been focused on their (limited) expressiveness. Morris et al. [42], Xu et al. [54] first proved that they are only as expressive as the Weisfeiler-Lehman test of isomorphism and Garg et al. [25] showed the limitations of basic directional message passing. Kondor et al. [34], Maron et al. [40], Morris et al. [42, 43] then investigated higher-order representations to circumvent this issue. Azizian & Lelarge [2], Maron et al. [39] then showed that so-called Folklore GNNs are the most expressive GNNs for a given tensor order.

Equivariant neural networks. Equivariance and invariance have recently emerged as one of the foundational principles of modern neural networks [11, 13]. This is especially relevant for models in physics, for which we often know the symmetries a priori. Equivariant models for the $\mathrm{SO}(3)$ group were first investigated in the context of spherical convolutions by Cohen et al. [12], Esteves et al. [19], Kondor et al. [33]. These methods leverage group representations to achieve full equivariance.

They were then transferred to the context of 3D point clouds and molecules by Anderson et al. [1], Thomas et al. [51], Weiler et al. [52], and further developed by Batzner et al. [4], Finzi et al. [22], Fuchs et al. [24]. Importantly, Yarotsky [55] proved the universality of 2D convolutional networks, and Bogatskiy et al. [5] extended this result to general groups. Maron et al. [41] proved universality for models invariant to S_n and equivariant to an additional symmetry. Dym & Maron [16] combined these results to prove the universality of models for the joined group of translations, rotations, and permutations, which is the exact group relevant for molecules.

3 Universality of spherical representations

GNNs for molecules typically incorporate directional information in one of two ways: Via $\mathrm{SO}(3)$ representations [1, 51]— whose universality was proved by Dym & Maron [16] — or by using directions in real space [32, 50]. Directions in real space are associated with the three-dimensional S^2 sphere, while the $\mathrm{SO}(3)$ group is double covered by the four-dimensional S^3 sphere. Directional representations thus use one degree of freedom less than $\mathrm{SO}(3)$ representations, making them cheaper but ultimately less expressive. Fortunately, in the special case of rotationally invariant predictions this lack of expressivity is only ostensible. We prove this by showing that in this case a model based on spherical representations is equivalent to the $\mathrm{SO}(3)$-based tensor field network (TFN) [51] variant used by Dym & Maron [16]. We can then relate this theorem to directional GNNs by interpreting them as a discretization of spherical representations.

Preliminaries. We consider a point cloud with n points (atoms), each associated with a position and a set of rotationally invariant features (e.g. atom types), defined as $\boldsymbol{X} \in \mathbb{R}^{3\times n}$ and $\boldsymbol{H}_{\text{in}} \in \mathbb{R}^{h\times n}$. In this section we define model classes by sets of functions $\mathcal{F}$. We are interested in proving that the set $\mathcal{F}$ defining our model is equal to the full set of functions $\mathcal{G}'$ that are invariant to the group of translations $\mathbb{T}^3$ and rotations $\mathrm{SO}(3)$, and equivariant to the group of permutations S_n. We denote the codomain of functions in $\mathcal{G}'$ as W_{T}^n, where W_{T} is some representation of $\mathrm{SO}(3)$. We denote a vector's norm by $x = \|\boldsymbol{x}\|_2$, its direction by $\hat{\boldsymbol{x}} = \boldsymbol{x}/x$, and the relative position by $\boldsymbol{x}_{ba} = \boldsymbol{x}_b - \boldsymbol{x}_a$. Proofs are deferred to the appendix.

Tensor field network. In order to show equivalence to spherical representations we start by defining TFN. Following Dym & Maron [16], we split the model into two parts: Embedding functions $\mathcal{F}_{\text{feat}}$ that lifts the input into an equivariant representation, and pooling functions $\mathcal{F}_{\text{pool}}$ that aggregate the results of multiple embedding functions on each point and computes the model output. The overall model is then defined as the set of functions

$$\mathcal{F}^{\text{TFN}}_{K(D),D} = \{f \mid f(\boldsymbol{X}, \boldsymbol{H}_{\text{in}}) = \sum_{k=1}^{K} f^{(k)*}_{\text{pool}}(f^{(k)}_{\text{feat}}(\boldsymbol{X}, \boldsymbol{H}_{\text{in}})), f^{(k)}_{\text{pool}} \in \mathcal{F}^{\text{TFN}}_{\text{pool}}(D), f^{(k)}_{\text{feat}} \in \mathcal{F}^{\text{TFN}}_{\text{feat}}(D)\}, \tag{1}$$

for some $K, D \in \mathbb{N}$, some general dependence $K(D)$, and with f^* denoting elementwise application of f on all points. The set $\mathcal{F}^{\text{TFN}}_{\text{pool}}$ contains all rotationally equivariant linear functions on the $\mathrm{SO}(3)$ group, i.e. all $\mathrm{SO}(3)$ convolutions [35]. Note that these are more expressive than the self-interaction layers used originally [51]. The embedding functions $\mathcal{F}^{\text{TFN}}_{\text{feat}}(D) = \{\pi_2 \circ f^{(2D)} \circ \cdots \circ f^{(1)} \mid f^{(i)} \in \mathcal{F}^{\text{TFN}}_{\text{prod}}\}$ consist of an auxiliary function $\pi_2(\boldsymbol{X}, \boldsymbol{H}) = \boldsymbol{H}$ and a series of tensor product functions (called convolution by Dym & Maron [16]) $\mathcal{F}^{\text{TFN}}_{\text{prod}} = \{f \mid f(\boldsymbol{X}, \boldsymbol{H}) = (\boldsymbol{X}, \tilde{\boldsymbol{H}}^{\text{TFN}}(\boldsymbol{X}, \boldsymbol{H}))\}$. The intermediate representations are $\boldsymbol{H} \in W^n_{\text{feat}}$, where W_{feat} is a representation of $\mathrm{SO}(3)$ indexed by the degree l and the order m ($l{=}m{=}0$ for $\boldsymbol{H}_{\text{in}}$). The main update is defined by

$$\tilde{\boldsymbol{H}}^{\text{TFN}(l_o)}_{am_o}(\boldsymbol{X}, \boldsymbol{H}) = \theta \boldsymbol{H}^{(l_o)}_{am_o} + \sum_{l_f, m_f} \sum_{l_i, m_i} C^{(l_o,m_o)}_{(l_f,m_f),(l_i,m_i)} \sum_{b \in \mathcal{N}_a} F^{(l_f)}_{\text{TFN},m_f}(\boldsymbol{x}_b - \boldsymbol{x}_a) \boldsymbol{H}^{(l_i)}_{bm_i}, \tag{2}$$

where θ is a (learned) scalar, $C^{(l_o,m_o)}_{(l_f,m_f),(l_i,m_i)}$ are the Clebsch-Gordan coefficients, and $\mathcal{N}_a$ are the neighbors of point a. $F^{(l)}_{\text{TFN},m}(\boldsymbol{x}) = R^{(l)}(x) Y_{lm}(\hat{\boldsymbol{x}})$ is a rotationally equivariant filter, with a (learned) radial part R, which is any polynomial of degree $\leq D$, and the real spherical harmonics Y_{lm}. The following universality result holds for this model:

Theorem 1 (Dym & Maron [16]). *Consider the set of functions $\mathcal{G}$ mapping $\mathbb{R}^{3\times n+h\times n} \to W_{\mathrm{T}}^n$ that are equivariant to rotations and permutations and invariant to translations. For all $n \in \mathbb{N}$,*

1. *For $D \in \mathbb{N}_0$, every polynomial $p \in \mathcal{G}$ of degree D is in $\mathcal{F}^{\text{TFN}}_{K(D),D}$.*
2. *Every continuous function $f \in \mathcal{G}$ can be approximated uniformly on compact sets by functions in $\bigcup_{D\in\mathbb{N}_0} \mathcal{F}^{\text{TFN}}_{K(D),D}$.*

Spherical networks. Instead of intermediate $\mathrm{SO}(3)$ representations we now switch to spherical representations, which are functions on the sphere $\boldsymbol{H} : S^2 \to \mathbb{R}$. In this case we define the set of functions $\mathcal{F}^{\text{sphere}}_{K(D),D}$ analogously to $\mathcal{F}^{\text{TFN}}_{K(D),D}$. However, for $\mathcal{F}^{\text{sphere}}_{\text{feat}}(D)$ we use

$$\tilde{\boldsymbol{H}}_a^{\text{sphere}}(\boldsymbol{X}, \boldsymbol{H})(\hat{\boldsymbol{r}}) = \theta \boldsymbol{H}_a(\hat{\boldsymbol{r}}) + \sum_{b\in\mathcal{N}_a} F_{\text{sphere}}(\boldsymbol{x}_b - \boldsymbol{x}_a, \hat{\boldsymbol{r}}) \boldsymbol{H}_b(\hat{\boldsymbol{r}}), \tag{3}$$

with the filter function $F_{\text{sphere}}(\boldsymbol{x}, \hat{\boldsymbol{r}}) = \sum_{l,m} R^{(l)}(x) \Re[Y_m^{(l)*}(\hat{\boldsymbol{x}}) Y_m^{(l)}(\hat{\boldsymbol{r}})]$, using the real part $\Re$ of the complex spherical harmonics $Y_m^{(l)}$. The set of pooling functions for invariant predictions is

$$\mathcal{F}^{\text{sphere}}_{\text{pool}} = \{f \mid f(\boldsymbol{H}) = \theta_{\text{pool}} \int_{S^2} \boldsymbol{H}(\hat{\boldsymbol{r}})\, \mathrm{d}\hat{\boldsymbol{r}}\}, \tag{4}$$

with the learnable parameter θ_{pool}. We obtain the universality theorem by showing the equivalence between this model and TFN for rotationally invariant functions. The proof is based on the connection between spherical harmonics and the Clebsch-Gordan coefficients [47, 3.7.72] (see App. A).

Theorem 2. *Consider the set of functions $\mathcal{G}'$ mapping $\mathbb{R}^{3\times n+h\times n} \to W_{\mathrm{T}}^n$ that are equivariant to permutations and invariant to translations and rotations. For all $n \in \mathbb{N}$,*

1. *For $D \in \mathbb{N}_0$, every polynomial $p \in \mathcal{G}'$ of degree D is in $\mathcal{F}^{\text{sphere}}_{K(D),D}$.*
2. *Every continuous function $f \in \mathcal{G}'$ can be approximated uniformly on compact sets by functions in $\bigcup_{D\in\mathbb{N}_0} \mathcal{F}^{\text{sphere}}_{K(D),D}$.*

Importantly, this theorem shows that we can approximate any rotationally invariant function using only representations on the S^2 sphere. We thus do not need $\mathrm{SO}(3)$ representations, spin-weighted spherical harmonics [20], triplet embeddings, or complex-valued functions. This result puts theory back in line with practice, where the best results are currently achieved without relying on these more expensive representations [45, 50].

4 From spherical representations to directional message passing

Directional representations. To use spherical representations in a model we first need to find a tractable description. Instead of using spherical harmonics, we propose to sample the representations in specific directions $\hat{\boldsymbol{r}}_i$. If we look at recent models, we see that they implicitly use the directions to each atom's neighbors for this purpose, i.e. they embed the edges in the molecule's graph. These directions define an *equivariant* mesh that circumvents the aliasing effects that would arise from fixed grids [33]. Schütt et al. [50] flexibly define the directional mesh in each layer by aggregating directions, while Klicpera et al. [32] and others use a fixed mesh for each atom. We can refine this mesh of directions e.g. by using more neighbors or by interpolating between directions. The approximation error of this directional mesh is related to the spherical harmonic expansion via the mesh norm and the separating distance between directions [28, 29]. Note that depending on the discretization scheme the resulting mesh might not provide a universal approximation guarantee.

Eq. (3) only defines the relationship for a fixed direction, while models commonly use different directional meshes for the input and output. To incorporate this we add a learned convolution, which only improves the model's expressiveness. Since the input and output are spherical functions, the

used filter F_2 has to be *zonal*, i.e. it can only depend on one angle. This can be expressed as [15]

$$\begin{aligned}\tilde{\boldsymbol{H}}_a^{\text{dir}}(\boldsymbol{X},\boldsymbol{H})(\hat{\boldsymbol{r}}_o) &= \theta \boldsymbol{H}_a(\hat{\boldsymbol{r}}_o) + \int_{\text{SO}(3)} \sum_{b\in\mathcal{N}_a} F_{\text{sphere}}(\boldsymbol{x}_{ba}, \boldsymbol{R}\hat{\boldsymbol{n}}) \sum_{i\in\mathcal{R}_b} \boldsymbol{H}_{bi}\delta(\boldsymbol{R}\hat{\boldsymbol{n}} - \hat{\boldsymbol{r}}_i) F_2(\boldsymbol{R}^{-1}\hat{\boldsymbol{r}}_o)\,\mathrm{d}\boldsymbol{R} \\ &= \theta \boldsymbol{H}_a(\hat{\boldsymbol{r}}_o) + \sum_{b\in\mathcal{N}_a}\sum_{i\in\mathcal{R}_b} F_{\text{sphere}}(\boldsymbol{x}_{ba}, \hat{\boldsymbol{r}}_i)\boldsymbol{H}_{bi} F_2(\measuredangle \hat{\boldsymbol{r}}_o\hat{\boldsymbol{r}}_i),\end{aligned} \tag{5}$$

where $\mathcal{R}_b$ denotes the directional mesh of atom b and the integral vanishes due to the Dirac delta δ.

General filters. To see the relationship to GNNs we furthermore need to generalize the filter $F_{\text{sphere}}(\boldsymbol{x}_{ba}, \hat{\boldsymbol{r}}_i)$. This filter only depends on the angle $\measuredangle \hat{\boldsymbol{r}}_i\hat{\boldsymbol{x}}_{ba}$ since it is rotationally invariant:

Lemma 1. $F_{\text{sphere}}(\boldsymbol{R}\boldsymbol{x}, \boldsymbol{R}\hat{\boldsymbol{r}}) = F_{\text{sphere}}(\boldsymbol{x}, \hat{\boldsymbol{r}})$ *for any rotation matrix* $\boldsymbol{R}$.

We can therefore substitute F_{sphere} with a general learnable filter F_1 that is parametrized by this relative angle. Since F_{sphere} arises as a special case we do not lose expressivity. We thus obtain

$$\tilde{\boldsymbol{H}}_a^{\text{gem}}(\boldsymbol{X},\boldsymbol{H})(\hat{\boldsymbol{r}}_o) = \theta \boldsymbol{H}_a(\hat{\boldsymbol{r}}_o) + \sum_{b\in\mathcal{N}_a}\sum_{i\in\mathcal{R}_b} F_1(x_{ba}, \measuredangle \hat{\boldsymbol{r}}_i\hat{\boldsymbol{x}}_{ba}) F_2(\measuredangle \hat{\boldsymbol{r}}_o\hat{\boldsymbol{r}}_i)\boldsymbol{H}_{bi}. \tag{6}$$

We have now arrived at a message passing scheme that has universal approximation guarantees and is only based on relative directional information. To see the connection to GNNs we interpret these discretized spherical representations as edge embeddings pointing towards $\hat{\boldsymbol{r}}_o$ and $\hat{\boldsymbol{r}}_i$. Eq. (6) then corresponds to two-hop message passing between the edge embeddings of $\hat{\boldsymbol{r}}_o$ and $\hat{\boldsymbol{r}}_i$ via the edge $\hat{\boldsymbol{x}}_{ba}$. Interestingly, the central learnable part of Eq. (6) is the product of the filters $F_1(x_{ba}, \measuredangle \hat{\boldsymbol{r}}_i\hat{\boldsymbol{x}}_{ba})$ and $F_2(\measuredangle \hat{\boldsymbol{r}}_o\hat{\boldsymbol{r}}_i)$ with the input representation, which is strikingly similar to the Hadamard product used in modern GNNs [31, 49] — except that these only use one-hop message passing.

5 Geometric message passing

Geometric representation. We now develop a specific two-hop message passing scheme based on Eq. (6). We use embeddings based on interatomic directions, and embed all atom pairs with distance $x_{ca} \leq c_{\text{emb}}$. $\hat{\boldsymbol{r}}_o$ and $\hat{\boldsymbol{r}}_i$ are thus instantiated as the interatomic directions $\hat{\boldsymbol{x}}_{ca}$ and $\hat{\boldsymbol{x}}_{db}$. We denote directional embeddings as $\boldsymbol{m}_{ca} = \boldsymbol{H}_a(\hat{\boldsymbol{x}}_{ca})$. Message passing is thus based on quadruplets of atoms — two atoms are interacting (a and b) and two atoms define the directions (c and d). We denote the angle between directions by $\varphi_{abd} = \measuredangle \hat{\boldsymbol{x}}_{ab}\hat{\boldsymbol{x}}_{db}$. To improve empirical performance we additionally use the dihedral angle $\theta_{cabd} = \measuredangle \hat{\boldsymbol{x}}_{ca}\hat{\boldsymbol{x}}_{db} \perp \hat{\boldsymbol{x}}_{ba}$ and substitute $\measuredangle \hat{\boldsymbol{r}}_o\hat{\boldsymbol{r}}_i = \measuredangle \hat{\boldsymbol{x}}_{ca}\hat{\boldsymbol{x}}_{db}$ with φ_{cab}. Fig. 1 illustrates the three angles φ_{cab}, φ_{abd}, and θ_{cabd} we use for updating the embedding $\boldsymbol{m}_{ca}$ based on $\boldsymbol{m}_{db}$. To ensure that all angles are well-defined we exclude overlapping atom quadruplets, i.e. $a \neq b \neq c \neq d$. We represent the relative directional information using spherical Fourier-Bessel bases with polynomial radial envelopes to ensure smoothly differentiable predictions, as proposed by Klicpera et al. [32]. We split the basis into three parts to incorporate all available geometric information. Before the envelope, these are:

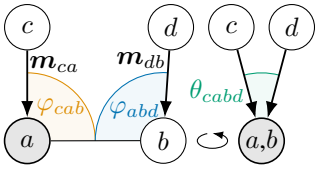

Figure 1: Angles used in geometric message passing.

$$\tilde{\boldsymbol{e}}_{\text{RBF},n}(x_{db}) = \sqrt{\frac{2}{c_{\text{emb}}}}\frac{\sin(\frac{n\pi}{c_{\text{emb}}}x_{db})}{x_{db}}, \tag{7}$$

$$\tilde{\boldsymbol{e}}_{\text{CBF},ln}(x_{ba}, \varphi_{abd}) = \sqrt{\frac{2}{c_{\text{int}}^3 j_{l+1}^2(z_{ln})}}\, j_l\!\left(\frac{z_{ln}}{c_{\text{int}}}x_{ba}\right) Y_0^{(l)}(\varphi_{abd}), \tag{8}$$

$$\tilde{\boldsymbol{e}}_{\text{SBF},lmn}(x_{ca}, \varphi_{cab}, \theta_{cabd}) = \sqrt{\frac{2}{c_{\text{emb}}^3 j_{l+1}^2(z_{ln})}}\, j_l\!\left(\frac{z_{ln}}{c_{\text{emb}}}x_{ca}\right) Y_m^{(l)}(\varphi_{cab}, \theta_{cabd}), \tag{9}$$

with the interaction cutoff c_{int}, the spherical Bessel functions j_l, and the n-th root of the l-order Bessel function z_{ln}. Note that Klicpera et al. [32] only used the first two parts $\boldsymbol{e}_{\text{RBF}}$ and $\boldsymbol{e}_{\text{CBF}}$.

These representations are then transformed using two linear layers to obtain the filter F (see Fig. 2 for details). In order to maintain a smoothly differentiable cutoff we cannot use a bias in this transformation. Altogether, the core geometric message passing scheme is

$$\tilde{\boldsymbol{m}}_{ca} = \sum_{\substack{b \in \mathcal{N}_a^{\text{int}} \setminus \{c\}, \\ d \in \mathcal{N}_b^{\text{emb}} \setminus \{a,c\}}} \Big((\boldsymbol{W}_{\text{SBF1}} \boldsymbol{e}_{\text{SBF}}(x_{ca}, \varphi_{cab}, \theta_{cabd}))^T \mathbf{W} ((\boldsymbol{W}_{\text{CBF2}} \boldsymbol{W}_{\text{CBF1}} \boldsymbol{e}_{\text{CBF}}(x_{ba}, \varphi_{abd})) \\ \odot (\boldsymbol{W}_{\text{RBF2}} \boldsymbol{W}_{\text{RBF1}} \boldsymbol{e}_{\text{RBF}}(x_{db})) \odot \boldsymbol{m}_{db}) \Big), \tag{10}$$

where $\boldsymbol{W}$ denotes a weight matrix, $\mathbf{W}$ denotes a weight tensor. The first weight matrix of each representation part has a small output dimension. This causes a bottleneck that improves generalization.

Symmetric message passing. Whenever we have a directional embedding $\boldsymbol{m}_{ca}$, we also have the opposing embedding $\boldsymbol{m}_{ac}$, since both are based on the same cutoff c_{emb}. Whether we associate the embedding $\boldsymbol{m}_{ca}$ or $\boldsymbol{m}_{ac}$ with atom a is arbitrary. A more principled approach is to interpret both embeddings as a *joint* representation of the atom pair a and c. In this view, an update to $\boldsymbol{m}_{ca}$ should also influence $\boldsymbol{m}_{ac}$. Normally, this would require executing the above message passing scheme twice. However, we can circumvent this overhead by performing the scheme only once and then selecting the relevant parts for $\boldsymbol{m}_{ac}$ and $\boldsymbol{m}_{ca}$ using learnable weight matrices. The message passing thus carries information for both embeddings jointly. Note that the embeddings still remain different, i.e. $\boldsymbol{m}_{ca} \neq \boldsymbol{m}_{ac}$. This only ensures better interaction between the two sides.

Efficient bilinear layer. The whole message passing scheme, i.e. basis transformation, neighbor aggregation, and bilinear layer only use linear functions. We can therefore freely optimize the order of summation without changing the result, as proposed by Wu et al. [53] (see App. C for details). Doing so can provide a faster and more memory-efficient model, reducing memory usage by $50\,\%$ even for Hadamard products. Moreover, since everything is based on efficient matrix products, this allows us to use the bilinear layer at practically no additional cost compared to a Hadamard product. Note that this requires using padded matrices instead of the usual gather-scatter operations to prevent excessively large intermediate results.

6 GemNet: Geometric message passing neural network

GemNet. The geometric message passing neural network (GemNet) is a significantly refined architecture based on DimeNet^{++} [31, Hippocratic license 2.1]. The architecture is illustrated in Fig. 2. A comprehensive version with low-level operations and hyperparameters is described in App. E. GemNet was developed on the COLL dataset, but generalizes to other datasets such as MD17 without architectural changes. Every change we propose either improves model performance or reduces model complexity. For example, GemNet only uses biases in the output layers since we found them to be irrelevant in other layers. We show the impact of the most relevant changes via ablation studies in Sec. 7.

Interactions. GemNet incorporates three forms of interactions. The first is geometric message passing, as described in Sec. 5. The second is a one-hop form of geometric message passing. This interaction uses a single cutoff $c = c_{\text{emb}}$ and passes messages only between directional embeddings pointing towards the same atom, similarly to DimeNet [32]. This provides both angle-based pair interactions and atom self-interactions, thanks to the symmetric message passing scheme described in Sec. 5. The third interaction is a pure atom self-interaction based on atom embeddings. We first aggregate the directional embeddings of one atom to obtain an atom embedding. We then use this atom embedding to update all directional embeddings. We found all three interaction forms to be beneficial, and show this in our ablation studies.

Stabilizing activation variance. The variance of activations in a model is usually stabilized using normalization methods, which has various positive effects on training [14, 38, 48]. However, they also have multiple undesirable side effects, especially in the context of molecular regression. Batch normalization introduces correlations between separate molecules and atoms. Layer normalization forces all activation scales to be constant, while atomic interactions actually cover a large range of scales — directly bonding atoms have a substantially stronger interaction than atoms at a long range. To circumvent these issues, we stabilize GemNet's variance by introducing constant scaling factors,

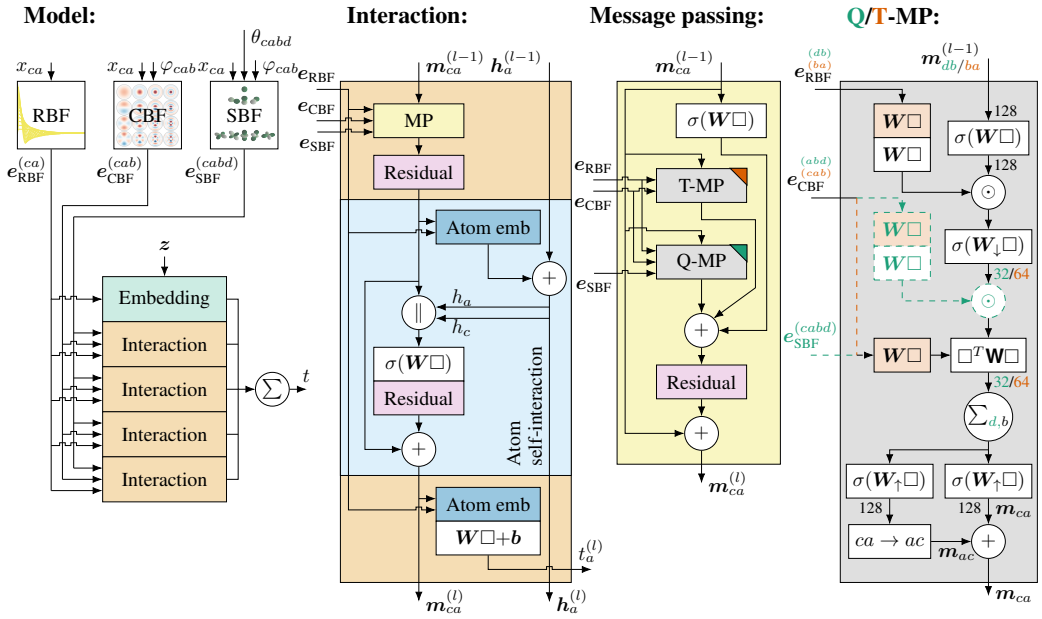

Figure 2: The GemNet architecture. $\square$ denotes the layer's input, $\|$ concatenation, and σ a non-linearity. Directional embeddings $\boldsymbol{m}_{ca}$ are updated using three forms of interaction: Two-hop geometric message passing (Q-MP), one-hop geometric message passing (T-MP), and atom self-interactions. Differences between Q-MP and T-MP are denoted by colors and dashed lines.

as suggested by Brock et al. [6]. We found that the activation variance is primarily impacted by four components: Skip connections, non-linearities, message aggregation, and Hadamard/bilinear layers. The two summands in a skip connection $y = x + f(x)$ have no covariance at initialization due to random weight matrices. We can thus remove its impact by scaling the output by $1/\sqrt{2}$. We remove the non-linearity's impact by scaling its output with a gain of $\gamma = 1/0.6$ for SiLU, similarly to [30]. Note that we do not center SiLU's output but instead choose a slightly lower γ to account for mean shift. Additionally, we standardize the weight matrices to have exactly zero mean and $1/\text{fan-in}$ variance. The sum aggregation and Hadamard/bilinear layers have a more complex impact on the variance, which we cannot determine a priori (see App. D for details). We therefore estimate the variance after these layers based on random batches of data. We then rescale their output accordingly to obtain roughly unit variance at initialization. These simple empirical scaling factors are sufficient to keep the activation variance roughly constant (see Fig. 3). We found that other measures such as adaptive gradient clipping [7], scaled weight standardization [6], or weighting the residual block with zero at initialization [14] are not beneficial for model accuracy.

GemNet-Q and GemNet-T. Geometric message passing is comparatively expensive since it is based on quadruplets of atoms. Its runtime thus scales with $\mathcal{O}(nk_{\text{int}}k_{\text{emb}}^2)$, where k_{int} is the number of interacting neighbors, and k_{emb} is the number of embedded directions. For this reason we investigate two message passing models in our experiments — one with two-hop geometric message passing (GemNet-Q) and one using only the two cheaper forms of interaction (GemNet-T), whose complexities are $\mathcal{O}(nk_{\text{emb}}^2)$ and $\mathcal{O}(nk_{\text{emb}})$. Note that GemNet-T is thus a direct ablation of the two-hop message passing scheme implied by our theoretical results.

Direct force predictions. GemNet predicts forces by calculating $\boldsymbol{F}_a = -\partial E / \partial \boldsymbol{x}_a$ via backpropagation. This form of calculation guarantees a conservative force field, which is important for the stability of simulations. However, GemNet can also directly predict forces and other vector quantities. We do this by predicting the magnitude of each directional embedding and then summing up over these vectors, similarly to Park et al. [44]. We denote this variant as *GemNet-dQ* and *GemNet-dT*. Interestingly, GemNet is thus able to generate rotationally *equivariant* predictions despite only using *invariant* representations. Direct predictions substantially accelerate the model, especially for training.

Table 1: MAE on COLL, in $\mathrm{meV}/\text{Å}$ and meV. Note that energies are only secondary for molecular dynamics.

	Forces	Energy
SchNet	172	198
DimeNet^{++}	40	**47**
GemNet-Q	**26.4**	53
GemNet-T	31.6	60
GemNet-dQ	38.1	60
GemNet-dT	43.1	55

Table 2: Force MAE for MD17@CCSD in $\mathrm{meV}/\text{Å}$. GemNet outperforms previous methods by $38\,\%$ on average.

	sGDML	NequIP	GemNet-Q	GemNet-T
Aspirin	33.0	14.7	11.5	**10.8**
Benzene	1.7	0.8	**0.7**	0.9
Ethanol	15.2	9.4	3.4	**3.3**
Malonaldehyde	16.0	16.0	6.5	**5.9**
Toluene	9.1	4.4	**2.9**	**2.9**

Table 3: Force MAE for MD17 in $\mathrm{meV}/\text{Å}$. GemNet outperforms all previous methods by a wide margin, on average by $38\,\%$.

	Kernel methods		GNNs				**GemNet**	
	sGDML	FCHL19	DimeNet	SphereNet	NequIP	PaiNN	GemNet-Q	GemNet-T
Aspirin	29.5	20.7	21.6	18.6	15.1	14.7	10.5	**10.0**
Benzene[8]	-	-	8.1	7.7	8.1	-	6.3	**6.1**
Benzene[9]	2.6	-	-	-	2.3	-	1.6	**1.4**
Ethanol	14.3	5.9	10.0	9.0	9.0	9.7	**3.8**	**3.8**
Malonaldehyde	17.8	10.6	16.6	14.7	14.6	14.9	7.2	**6.8**
Naphthalene	4.8	6.5	9.3	7.7	4.2	3.3	**2.3**	2.6
Salicylic acid	12.1	9.6	16.2	15.6	10.3	8.5	5.7	**5.6**
Toluene	6.1	8.8	9.4	6.7	4.4	4.1	2.8	**2.7**
Uracil	10.4	4.6	13.1	11.6	7.5	6.0	4.5	**4.2**

The resulting accuracy is on par with most previous models, but significantly worse than GemNet's accuracy via backpropagation. Note that Theorem 2 does not cover equivariant vector predictions.

Limitations. GemNet is focused on one specific, important task: Predictions for molecular simulations. We do not make any statements regarding its performance beyond this scope. The GemNet architecture might seem more complex than some previous models, due to its larger variety of interactions and blocks. However, its number of parameters and training or inference time is actually on par with previous models. Two-hop message passing introduces significant computational overhead. We mitigate this effect with a down-projection layer and additionally introduce the ablated GemNet-T model. This model performs surprisingly well on MD17, but not on COLL. This suggests that one-hop message passing is expressive enough for some practical use cases, but two-hop message passing gives an advantage for the more challenging task of fitting multiple molecules at once.

Societal impacts. Accelerating molecular simulations can have positive effects in a wide range of applications in physics and chemistry. At the same time, however, this can be used for malicious purposes such as developing chemical agents or weapons. To the best of our knowledge, this work does not promote these use cases more than regular chemistry research does. To somewhat mitigate negative effects we will publish our source code under the Hippocratic license [17].

7 Experiments

Experimental setup. We evaluate our model on three molecular dynamics datasets. COLL [31, CC-BY 4.0] consists of configurations taken from molecular collisions of different small organic molecules. MD17 [8] consists of configurations of multiple separate, thermalized molecules, considering only one molecule at a time. MD17@CCSD [9] uses the same setup, but calculates the forces using the more accurate and expensive CCSD or CCSD(T) method. Following the setup of Batzner et al. [4], we use 1000 training and validation configurations for MD17, and 950 training and 50 validation configurations for MD17@CCSD. We focus on force predictions and use a high force loss weight since they determine the accuracy of molecular simulations. We measure the mean absolute error (MAE), averaged over all samples, atoms, and components. We compare with the

results reported by several state-of-the-art models: sGDML [9], FCHL19 [10], SchNet [49], DimeNet [32], DimeNet^{++} [31], SphereNet [36], NequIP [4], and PaiNN [50]. For further details see App. F.

Results. Tables 1 to 3 show that GemNet-T and GemNet-Q consistently perform best on all molecular dynamics datasets investigated — and by a large margin. This is true both in comparison to previous GNNs and for kernel methods — despite the latter typically being more sample efficient. The improvements are largest for chain-like molecules, such as ethanol and malonaldehyde. These molecules are the most challenging since they exhibit a wide range of movement. GemNet even performs better than some previous models that were trained with 50x more training samples. For example, it performs better than SchNet with $50\,000$ training samples on six out of eight MD17 molecules (see Table 8). Interestingly, the two-hop message passing scheme implied by our theoretical results (GemNet-Q) yields significant improvements on COLL, but performs worse than the ablated GemNet-T on MD17. COLL covers a much larger variety of configurations and molecules. This result thus suggests that two-hop message passing is especially important for generalization in more difficult settings that cover large regions of chemical space.

Computational aspects. GemNet-Q is roughly two times slower than GemNet-T for training, but equally fast for inference (see Table 9). Thanks to efficient aggregation, GemNet with bilinear layers is as fast as with regular Hadamard products. Efficient aggregation also reduces the memory usage for regular Hadamard products by around $50\,\%$ (from 4.1GB to 2.2GB for a batch of 32 Toluene molecules). Note that GemNet has not been optimized for runtime and can likely be substantially accelerated. GemNet-Q uses 2.2M and GemNet-T 1.9M parameters, which is comparable to previous models such as DimeNet^{++}, which uses 1.9M parameters. See App. H for further details.

Direct force prediction. Directly predicting the forces accelerates training by up to three times, but has little impact on inference speed in our experiments (see Table 9). This can likely be attributed to the GPU not being fully utilized. However, direct force prediction has a strong detrimental effect on accuracy. Using backpropagation reduces the MAE by $44\,\%$ on COLL and by $52\,\%$ on MD17 (see Tables 1 and 6). Note that the resulting MAE is still better than most previous methods. Whether to use direct predictions thus depends on the application's computational requirements.

Ablation studies. We investigate the proposed architectural improvements on COLL in Table 4. The proposed symmetric message passing scheme yields significant accuracy improvements, as does using a bilinear layers instead of a Hadamard product. We also see that removing any of the three interaction forms described in Sec. 6 increases the error, showing that this combination is indeed beneficial. The proposed scaling factors also yield decent improvements, while regular layer normalization actually increases the error. Two-hop message passing yields the largest single improvement. However, the error improvements are quite evenly distributed overall. This suggests that GemNet's improved performance is not due to one single change, but rather due to the full range of improvements proposed in this work.

Table 4: Ablation studies on COLL. Force MAE in $\mathrm{meV/Å}$ after $500\,000$ training steps. All proposed components yield significant improvements.

Model	Forces
without symmetric message passing	28.5
Hadamard product instead of bilinear layer	29.3
without atom embedding updates	28.3
without one-hop message passing	31.3
without two-hop message passing	32.4
without scaling factors	29.1
use layer norm instead (without centering)	33.3
with bias	27.2
GemNet-Q	27.0

8 Conclusion

In this work we proved the universality for GNNs using directional embeddings. We proposed geometric message passing based on these insights, and improved this scheme with symmetric message passing and efficient bilinear layers. We incorporated these improvements in the GemNet architecture, which substantially improves the error on various molecular dynamics datasets. We showed that all of the proposed enhancements yield significant performance improvements. Most of these these are of independent interest for other molecular GNNs.

References

[1] Brandon M. Anderson, Truong-Son Hy, and Risi Kondor. Cormorant: Covariant Molecular Neural Networks. In *NeurIPS*, 2019.

[2] Waiss Azizian and Marc Lelarge. Expressive Power of Invariant and Equivariant Graph Neural Networks. In *ICLR*, 2020.

[3] Albert P. Bartók, Mike C. Payne, Risi Kondor, and Gábor Csányi. Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons. *Physical Review Letters*, 104(13):136403, 2010.

[4] Simon Batzner, Tess E. Smidt, Lixin Sun, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, and Boris Kozinsky. SE(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials. *arXiv*, 2101.03164, 2021.

[5] Alexander Bogatskiy, Brandon Anderson, Jan Offermann, Marwah Roussi, David Miller, and Risi Kondor. Lorentz Group Equivariant Neural Network for Particle Physics. In *ICML*, 2020.

[6] Andrew Brock, Soham De, and Samuel L. Smith. Characterizing signal propagation to close the performance gap in unnormalized ResNets. In *ICLR*, 2021.

[7] Andrew Brock, Soham De, Samuel L. Smith, and Karen Simonyan. High-Performance Large-Scale Image Recognition Without Normalization. *arXiv*, 2102.06171, 2021.

[8] 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.

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

[10] Anders S. Christensen, Lars A. Bratholm, Felix A. Faber, and O. Anatole von Lilienfeld. FCHL revisited: Faster and more accurate quantum machine learning. *The Journal of Chemical Physics*, 152(4):044107, 2020.

[11] Taco Cohen and Max Welling. Group Equivariant Convolutional Networks. In *ICML*, 2016.

[12] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *ICLR*, 2018.

[13] Taco S Cohen, Mario Geiger, and Maurice Weiler. A General Theory of Equivariant CNNs on Homogeneous Spaces. In *NeurIPS*, 2019.

[14] Soham De and Sam Smith. Batch Normalization Biases Residual Blocks Towards the Identity Function in Deep Networks. In *NeurIPS*, 2020.

[15] J. R. Driscoll and D. M. Healy. Computing Fourier Transforms and Convolutions on the 2-Sphere. *Advances in Applied Mathematics*, 15(2):202–250, 1994.

[16] Nadav Dym and Haggai Maron. On the Universality of Rotation Equivariant Point Cloud Networks. In *ICLR*, 2021.

[17] Coraline Ada Ehmke. The Hippocratic License 2.1: An Ethical License for Open Source., 2020. URL `https://firstdonoharm.dev`.

[18] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018.

[19] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning SO(3) Equivariant Representations with Spherical CNNs. In *ECCV*, 2018.

[20] Carlos Esteves, Ameesh Makadia, and Kostas Daniilidis. Spin-Weighted Spherical CNNs. In *NeurIPS*, 2020.

[21] Felix A. Faber, Anders S. Christensen, Bing Huang, and O. Anatole von Lilienfeld. Alchemical and structural distribution based representation for universal quantum machine learning. *The Journal of Chemical Physics*, 148(24):241717, 2018.

[22] Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing Convolutional Neural Networks for Equivariance to Lie Groups on Arbitrary Continuous Data. In *ICML*, 2020.

[23] Daniel Flam-Shepherd, Tony C. Wu, Pascal Friederich, and Alan Aspuru-Guzik. Neural Message Passing on High Order Paths. *Machine Learning: Science and Technology*, 2021.

[24] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks. In *NeurIPS*, 2020.

[25] Vikas Garg, Stefanie Jegelka, and Tommi Jaakkola. Generalization and Representational Limits of Graph Neural Networks. In *ICML*, 2020.

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

[27] Thomas A. Halgren. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry*, 17(5-6):490–519, 1996.

[28] Kurt Jetter, Joachim Stöckler, and Joseph Ward. Error estimates for scattered data interpolation on spheres. *Mathematics of Computation*, 68(226):733–747, 1999.

[29] Jens Keiner, Stefan Kunis, and Daniel Potts. Efficient Reconstruction of Functions on the Sphere from Scattered Data. *Journal of Fourier Analysis and Applications*, 13(4):435–458, 2007.

[30] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. In *NeurIPS*, 2017.

[31] Johannes Klicpera, Shankari Giri, Johannes T. Margraf, and Stephan Günnemann. Fast and Uncertainty-Aware Directional Message Passing for Non-Equilibrium Molecules. In *NeurIPS-W*, 2020.

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

[33] Risi Kondor, Zhen Lin, and Shubhendu Trivedi. Clebsch-Gordan Nets: a Fully Fourier Space Spherical Convolutional Neural Network. In *NeurIPS*, 2018.

[34] Risi Kondor, Truong Son Hy, Horace Pan, Brandon M. Anderson, and Shubhendu Trivedi. Covariant Compositional Networks For Learning Graphs. In *ICLR-W*, 2019.

[35] Peter J. Kostelec and Daniel N. Rockmore. FFTs on the Rotation Group. *Journal of Fourier Analysis and Applications*, 14(2):145–179, 2008.

[36] Yi Liu, Limei Wang, Meng Liu, Xuan Zhang, Bora Oztekin, and Shuiwang Ji. Spherical Message Passing for 3D Graph Networks. *arXiv*, 2102.05013, 2021.

[37] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *ICLR*, 2018.

[38] Ping Luo, Xinjiang Wang, Wenqi Shao, and Zhanglin Peng. Towards Understanding Regularization in Batch Normalization. In *ICLR*, 2019.

[39] Haggai Maron, Heli Ben-Hamu, Hadar Serviansky, and Yaron Lipman. Provably Powerful Graph Networks. In *NeurIPS*, 2019.

[40] Haggai Maron, Ethan Fetaya, Nimrod Segol, and Yaron Lipman. On the Universality of Invariant Networks. In *ICML*, 2019.

[41] Haggai Maron, Or Litany, Gal Chechik, and Ethan Fetaya. On Learning Sets of Symmetric Elements. In *ICML*, 2020.

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

[43] Christopher Morris, Gaurav Rattan, and Petra Mutzel. Weisfeiler and Leman go sparse: Towards scalable higher-order graph embeddings. In *NeurIPS*, 2020.

[44] Cheol Woo Park, Mordechai Kornbluth, Jonathan Vandermause, Chris Wolverton, Boris Kozinsky, and Jonathan P. Mailoa. Accurate and scalable multi-element graph neural network force field and molecular dynamics with direct force architecture. *arXiv*, 2007.14444, 2020.

[45] Zhuoran Qiao, Matthew Welborn, Animashree Anandkumar, Frederick R. Manby, and Thomas F. Miller. OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features. *The Journal of Chemical Physics*, 153(12):124111, 2020.

[46] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the Convergence of Adam and Beyond. In *ICLR*, 2018.

[47] J. J. Sakurai and San Fu Tuan. *Modern Quantum Mechanics*. Revised, subsequent edition edition, 1993.

[48] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Mądry. How does batch normalization help optimization? In *NeurIPS*, 2018.

[49] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *NeurIPS*, 2017.

[50] Kristof T. Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. *arXiv*, 2102.03150, 2021.

[51] Nathaniel Thomas, Tess Smidt, Steven M. Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor Field Networks: Rotation- and Translation-Equivariant Neural Networks for 3D Point Clouds. *arXiv*, 1802.08219, 2018.

[52] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco Cohen. 3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data. In *NeurIPS*, 2018.

[53] Wenxuan Wu, Zhongang Qi, and Li Fuxin. PointConv: Deep Convolutional Networks on 3D Point Clouds. In *CVPR*, 2019.

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

[55] Dmitry Yarotsky. Universal Approximations of Invariant Maps by Neural Networks. *Constructive Approximation*, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Sec. 6.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See end of Sec. 6.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] As is common molecular dynamics models, our results are quite stable w.r.t. random seeds and even data splits, so we did not run them multiple times.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No] We were not able to find a license for MD17.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]