

# CRYSTALFORMER: INFINITELY CONNECTED ATTENTION FOR PERIODIC STRUCTURE ENCODING

Tatsunori Taniai<sup>1</sup>, Ryo Igarashi<sup>1</sup>, Yuta Suzuki<sup>2</sup>, Naoya Chiba<sup>3</sup>, Kotaro Saito<sup>4,5</sup>  
Yoshitaka Ushiku<sup>1</sup>, Kanta Ono<sup>5</sup>

<sup>1</sup>OMRON SINIC X Corp., <sup>2</sup>Toyota Motor Corp., <sup>3</sup>Tohoku University, <sup>4</sup>Randeft Inc., <sup>5</sup>Osaka University  
<https://omron-sinix.github.io/crystalformer/>

## ABSTRACT

Predicting physical properties of materials from their crystal structures is a fundamental problem in materials science. In peripheral areas such as the prediction of molecular properties, fully connected attention networks have been shown to be successful. However, unlike these finite atom arrangements, crystal structures are infinitely repeating, periodic arrangements of atoms, whose fully connected attention results in *infinitely connected attention*. In this work, we show that this infinitely connected attention can lead to a computationally tractable formulation, interpreted as *neural potential summation*, that performs infinite interatomic potential summations in a deeply learned feature space. We then propose a simple yet effective Transformer-based encoder architecture for crystal structures called *Crystalformer*. Compared to an existing Transformer-based model, the proposed model requires only 29.4% of the number of parameters, with minimal modifications to the original Transformer architecture. Despite the architectural simplicity, the proposed method outperforms state-of-the-art methods for various property regression tasks on the Materials Project and JARVIS-DFT datasets.

## 1 INTRODUCTION

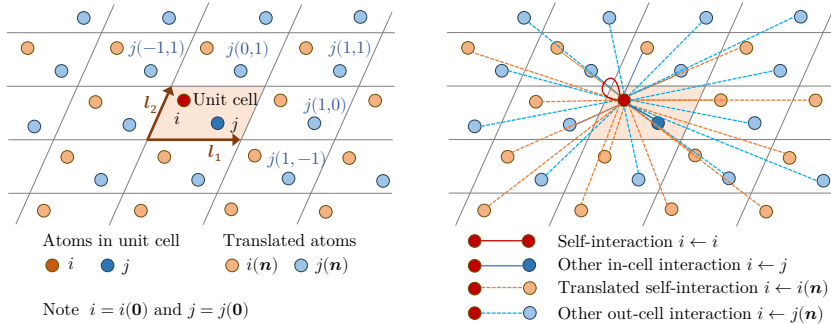
Predicting physical properties of materials from their crystal structures without actually synthesizing materials is important for accelerating the discovery of new materials with desired properties (Pollice et al., 2021). While physical simulation methods such as density functional theory (DFT) calculations can accurately simulate such properties, their high computational load largely limits their applicability, for example, in large-scale screening of potentially valuable materials. Thus, high-throughput machine learning (ML)-based approaches are actively studied (Pollice et al., 2021; Choudhary et al., 2022).

Since crystal structures and molecules are both 3D arrangements of atoms, they share similar challenges in their encoding for property prediction, such as permutation invariance and SE(3) invariance (i.e., rotation and translation invariance). Hence, similar approaches using graph neural networks (GNNs) are popular for invariantly encoding these 3D structures (Reiser et al., 2022).

Against this mainstream of GNN variants, approaches based on Transformer encoders (Vaswani et al., 2017) are emerging recently and showing superior performance in property prediction of molecules (Ying et al., 2021; Liao & Smidt, 2023) and crystal structures (Yan et al., 2022). Particularly, Graphormer by Ying et al. (2021) adopts fully connected attention between atoms in a molecule, by following the standard Transformer architecture, and showed excellent prediction performance.

Crystal structures, however, have a unique structural feature —periodicity— that produces infinitely repeating periodic arrangements of atoms in 3D space. Because of the periodicity, fully connected attention between atoms in a crystal structure leads to a non-trivial formulation, namely *infinitely connected attention* (see Fig. 1), which involves infinite series over repeated atoms. The previous method, called Matformer (Yan et al., 2022), avoids such a formulation and presents itself rather as a hybrid of Transformer and message passing GNN. Thus, whether the standard Transformer architecture is effectively applicable to crystal structure encoding is still an open question.

In this work, we interpret this infinitely connected attention as a physics-inspired infinite summation of interatomic potentials performed deeply in abstract feature space, which we call *neural potential*

Figure 1: **2D diagrams of crystal structure and infinitely connected attention.**

*summation*. In this view, attention weights are formulated as interatomic distance-decay potentials, which make the infinitely connected attention approximately tractable. By using this formulation, we propose a simple yet effective Transformer encoder for crystal structures called *Crystalformer*, establishing a novel Transformer framework for periodic structure encoding. Compared to the previous work by Yan et al. (2022), we aim to develop the formalism for more faithful Transformer-based crystal structure encoding. The resulting architecture is shown to require only 29.4% of the total number of parameters of their Matformer to achieve better performance, while sharing useful invariance properties. We also point out that a direct extension of Graphormer (Ying et al., 2021) for periodic structures leads to inadequate modeling of periodicity, which the proposed framework overcomes. We further show that the proposed framework using the infinitely connected attention formulation is beneficial for efficiently incorporating long-range interatomic interactions.

Quantitative comparisons using Materials Project and JARVIS-DFT datasets show that the proposed method outperforms several neural-network-based state-of-the-art methods (Xie & Grossman, 2018; Schütt et al., 2018; Chen et al., 2019; Louis et al., 2020; Chen & Ong, 2022; Choudhary & DeCost, 2021; Yan et al., 2022) for various crystal property prediction tasks. We release our code online.

## 2 PRELIMINARIES

We begin with introducing the unit cell representation of crystal structures and recapping standard self-attention in Transformer encoders with relative position representations.

### 2.1 CRYSTAL STRUCTURES

Assume a periodic structure system on a lattice in 3D space, which consists of finite sets of points and their attributes,  $P = \{p_1, p_2, \dots, p_N\}$  and  $X^0 = \{x_1^0, x_2^0, \dots, x_N^0\}$ , in a unit cell as well as lattice vectors,  $l_1, l_2, l_3 \in \mathbb{R}^3$ , defining the unit cell translations. As a crystal structure, each  $p_i \in \mathbb{R}^3$  and  $x_i^0 \in \mathbb{R}^d$  represent the Cartesian coordinate and species of an atom in the unit cell, respectively. Any point in the system is then located at a translated position, or an *image*, of a unit-cell point as

$$p_{i(n)} = p_i + n_1 l_1 + n_2 l_2 + n_3 l_3, \quad (1)$$

where three integers in  $n = (n_1, n_2, n_3) \in \mathbb{Z}^3$  define a 3D translation with  $l_1, l_2, l_3$ . See Fig. 1 (left) for an illustration. For brevity, let  $i$  denote the index of the  $i$ -th unit-cell point without translation, and let  $i(n)$  denote the index of a translated image of  $i$  (including  $i$  itself as  $i(0)$ ).  $\sum_n$  denotes the infinite series over  $\mathbb{Z}^3$ . Indices  $j$  and  $j(n)$  are used similarly.

### 2.2 SELF-ATTENTION WITH RELATIVE POSITIONS

Self-attention (Vaswani et al., 2017) with relative position representations (Shaw et al., 2018) transforms an ordered finite sequence of feature vectors to another, as  $(x_1, \dots, x_N) \rightarrow (y_1, \dots, y_N)$ :

$$y_i = \frac{1}{Z_i} \sum_{j=1}^N \exp(q_i^T k_j / \sqrt{d_K} + \phi_{ij}) (v_j + \psi_{ij}), \quad (2)$$

where query  $\mathbf{q}$ , key  $\mathbf{k}$ , value  $\mathbf{v}$  are linear projections of input  $\mathbf{x}$ ,  $Z_i = \sum_{j=1}^N \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \phi_{ij})$  is the normalizer of softmax attention weights, and  $d_K$  is the dimensionality of  $\mathbf{k}$  and  $\mathbf{q}$ . Since each input  $\mathbf{x}_i$  is position-agnostic and so are pairwise similarity  $\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K}$  and value  $\mathbf{v}_j$ , they are augmented with scalar  $\phi_{ij}$  and vector  $\boldsymbol{\psi}_{ij}$  biases that encode/embed the relative position,  $j - i$ .

### 3 CRYSTALFORMER

We consider a problem of estimating physical properties of a given crystal structure. Following Xie & Grossman (2018) and Schütt et al. (2018), we represent a physical state of the whole structure as a finite set of abstract state variables for the atoms in the unit cell,  $X = f\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N g$ , assuming that any atom in the structure shares the state with its corresponding unit-cell atom, as  $\mathbf{x}_{i(n)} = \mathbf{x}_i$ . Given input state  $X^0$  that only symbolically represents the species of the unit-cell atoms, we evolve state  $X^0$ , through repeated interactions between atom-wise states, to another  $X^0$  reflecting the target properties for prediction. To this end, we propose an attention operation for state  $X$ , which is induced by the input structure specified with the unit cell points  $f\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_N g$  and lattice vectors  $f\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3 g$ .

#### 3.1 INFINITELY CONNECTED ATTENTION AS NEURAL POTENTIAL SUMMATION

Inspired by Graphormer (Ying et al., 2021)’s fully connected attention for atoms in a molecule, we formulate similar dense attention for a crystal structure for its state evolution. Compared to finite graph representations in GNNs, this dense formulation more faithfully represents the physical phenomena occurring inside crystal structures and should provide a good starting point for discussion. Because of the crystal periodicity, such attention amounts to pairwise interactions between the unit-cell atoms  $i$  as queries and all the infinitely repeated atoms  $j(n)$  in the structure as keys/values as

$$\mathbf{y}_i = \frac{1}{Z_i} \sum_{j=1}^N \sum_{\mathbf{n}} \exp\left(\mathbf{q}_i^T \mathbf{k}_{j(n)} / \sqrt{d_K} + \phi_{ij(n)}\right) (\mathbf{v}_{j(n)} + \boldsymbol{\psi}_{ij(n)}), \quad (3)$$

where  $Z_i = \sum_{j=1}^N \sum_{\mathbf{n}} \exp(\mathbf{q}_i^T \mathbf{k}_{j(n)} / \sqrt{d_K} + \phi_{ij(n)})$ . Notice  $\mathbf{k}_{j(n)} = \mathbf{k}_j$  and  $\mathbf{v}_{j(n)} = \mathbf{v}_j$  since  $\mathbf{x}_{j(n)} = \mathbf{x}_j$ . Fig. 1 (right) illustrates these dense connections. We regard this infinitely connected attention in Eq. (3) as a *neural potential summation*. In physical simulations, energy calculations typically involve infinite summation  $\sum_{j(n) \neq i} (k\mathbf{p}_{j(n)} - \mathbf{p}_i k)v_{j(n)}$  for potential function  $(r)$  and physical quantity  $v$  (e.g., electric charge). Analogously, Eq. (3) computes the state of each unit-cell atom,  $\mathbf{y}_i$ , by summing abstract influences,  $(\mathbf{v}_{j(n)} + \boldsymbol{\psi}_{ij(n)})$ , from all the atoms in the structure,  $j(n)$ , with their weights provided by abstract interatomic scalar potentials,  $\exp(\mathbf{q}_i^T \mathbf{k}_{j(n)} / \sqrt{d_K} + \phi_{ij(n)}) / Z_i$ . Since  $\mathbf{q}_i^T \mathbf{k}_{j(n)} / \sqrt{d_K}$  and  $\mathbf{v}_{j(n)}$  are position-agnostic, they are augmented with relative position encodings,  $\phi_{ij(n)}$  and  $\boldsymbol{\psi}_{ij(n)}$ , to reflect the interatomic spatial relation (i.e.,  $\mathbf{p}_{j(n)} - \mathbf{p}_i$ ). Thus,  $\exp(\phi_{ij(n)})$  is interpreted as a spatial dependency factor of the interatomic potential between  $i$  and  $j(n)$ , and  $\boldsymbol{\psi}_{ij(n)}$  as an abstract position-dependent influence on  $i$  from  $j(n)$ .

Compared to the standard finite-element self-attention in Eq. (2), the main challenge in computing Eq. (3) is the presence of infinite series  $\sum_{\mathbf{n}}$  owing to the crystal periodicity.

**Pseudo-finite periodic attention.** With simple algebra (see Appendix A), we can rewrite Eq. (3) as

$$\mathbf{y}_i = \frac{1}{Z_i} \sum_{j=1}^N \exp\left(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \alpha_{ij}\right) (\mathbf{v}_j + \boldsymbol{\beta}_{ij}), \quad (4)$$

where

$$\alpha_{ij} = \log \sum_{\mathbf{n}} \exp(\phi_{ij(n)}), \quad (5)$$

$$\boldsymbol{\beta}_{ij} = \frac{1}{Z_{ij}} \sum_{\mathbf{n}} \exp(\phi_{ij(n)}) \boldsymbol{\psi}_{ij(n)}, \quad (6)$$

and  $Z_{ij} = \sum_{\mathbf{n}} \exp(\phi_{ij(n)})$ . Eq. (4) now resembles the standard finite-element attention in Eq. (2), if  $\alpha_{ij}$  and  $\boldsymbol{\beta}_{ij}$  are viewed as quantities encoding the relative position between  $i$  and  $j$  in finite element set  $f1, 2, \dots, N g$ . In reality,  $\alpha_{ij}$  is the log-sum of spatial dependencies  $\exp(\phi_{ij(n)})$  between  $i$  and the  $j$ ’s

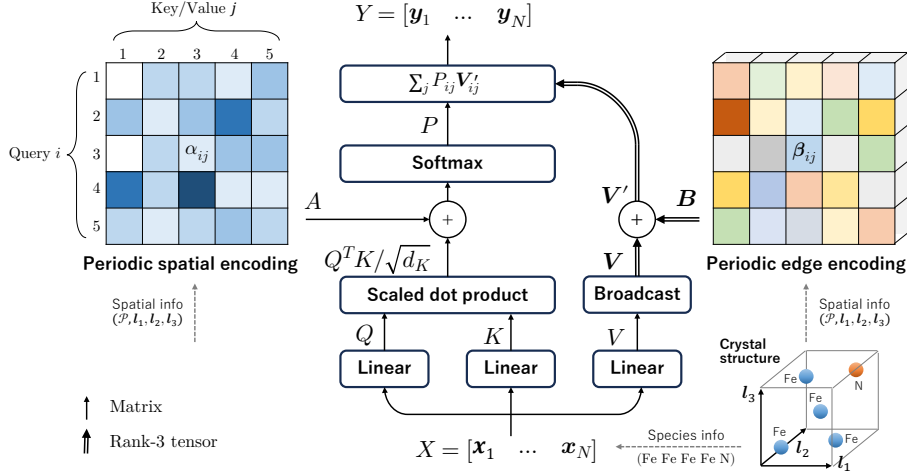


Figure 2: **Pseudo-finite periodic attention with periodic spatial and edge encoding in a matrix-tensor diagram.** Scalar  $\alpha_{ij}$  and vector  $\beta_{ij}$  integrate the spatial relations between unit-cell atom  $i$  and the  $j$ 's all repeated atoms, allowing the infinitely connected attention to be performed as standard fully connected attention for finite unit-cell atoms. (Unlike usual notation,  $X, Q, K, V, Y$  here denote column-vector-based feature matrices for better consistency with the notation in the main text.)

all images. Likewise,  $\beta_{ij}$  is the softmax-weighted average of position-dependent influences  $\psi_{ij(n)}$  on  $i$  from the  $j$ 's all images, weighted by their spatial dependencies. We call  $\alpha_{ij}$  and  $\beta_{ij}$  the *periodic spatial encoding* and *periodic edge encoding*, respectively, and call the infinitely connected attention with these encodings the *pseudo-finite periodic attention*. If  $\alpha_{ij}$  and  $\beta_{ij}$  are tractably computed, this attention can be performed similarly to the standard finite-element attention, as shown in Fig. 2.

**Distance decay attention.** In the physical world, spatial dependencies between two atoms tend to decrease as their distance increases. To approximate such dependencies with  $\exp(\phi_{ij(n)})$ , we adopt the following Gaussian distance decay function as a simple choice among other possible forms.

$$\exp(\phi_{ij(n)}) = \exp\left(-\frac{k\mathbf{p}_j(n) \cdot \mathbf{p}_i k^2 / 2\sigma_i^2}{k\mathbf{p}_j(n) \cdot \mathbf{p}_i k^2 / 2\sigma_i^2}\right) \quad (7)$$

Here,  $\sigma_i > 0$  is a scalar variable controlling the Gaussian tail length. When  $\sigma_i$  is not too large, series  $\sum_n$  in Eqs. (5) and (6) converges quickly as  $k\mathbf{n}k$  grows with a provable error bound (see Appendix B), thus making  $\alpha_{ij}$  and  $\beta_{ij}$  computationally tractable. To ensure the tractability, we model  $\sigma_i$  as a function of current state  $\mathbf{q}_i$  with a fixed upper bound, as  $\sigma_i < \sigma_{ub}$  (see Appendix C for the complete definition of  $\sigma_i$ ). We found that, although allowing relatively long tails (e.g.,  $\sigma_{ub} = 7\text{\AA}$ ) is still tractable, the use of a shorter tail bound (e.g.,  $2\text{\AA}$ ) empirically leads to better results. The methodology to eliminate the bound for  $\sigma_i$  is further discussed in Sec. 6.

**Value position encoding for periodicity-aware modeling.** Value position encoding  $\psi_{ij(n)}$  in Eq. (3) represents position-dependent influences on  $i$  from  $j(n)$ , and is another key to properly encoding periodic structures. In fact, attention without  $\psi_{ij(n)}$  cannot distinguish between crystal structures consisting of the same single unit-cell atom with different lattice vectors. This is obvious since Eq. (3) with  $\psi_{ij(n)} = \mathbf{0}$  and  $N = 1$  degenerates to  $\mathbf{y}_1 = \mathbf{v}_1$ , which is completely insensible to the input lattice vectors (see Appendix D for more discussions). As a simple choice for  $\psi_{ij(n)}$ , we borrow edge features used by existing GNNs (Schütt et al., 2017; Xie & Grossman, 2018). These edge features are Gaussian radial basis functions  $\mathbf{b}(r) = (b_1, b_2, \dots, b_K)^T$  defined as

$$b_k(r) = \exp\left(-\frac{(r - \mu_k)^2 / 2(r_{\max}/K)^2}{(r - \mu_k)^2 / 2(r_{\max}/K)^2}\right), \quad (8)$$

where  $\mu_k = kr_{\max}/K$ , and  $K$  and  $r_{\max}$  are hyperparameters. Intuitively, vector  $\mathbf{b}(r)$  quantizes scalar distance  $r$  via soft one-hot encoding using  $K$  bins equally spaced between 0 and  $r_{\max}$ . We provide  $\psi_{ij(n)}$  as a linear projection of  $\mathbf{b}(r)$  with trainable weight matrix  $W^E$  as

$$\psi_{ij(n)} = W^E \mathbf{b}(k\mathbf{p}_j(n) \cdot \mathbf{p}_i k). \quad (9)$$

**Implementation details.** When computing  $\alpha_{ij}$  and  $\beta_{ij}$  in Eqs. (5) and (6), the Gaussian functions in series  $\sum_n$  mostly decays rapidly within a relatively small range of  $k\mathbf{n}k_1 \approx 2$  (i.e., supercell of  $5^3$

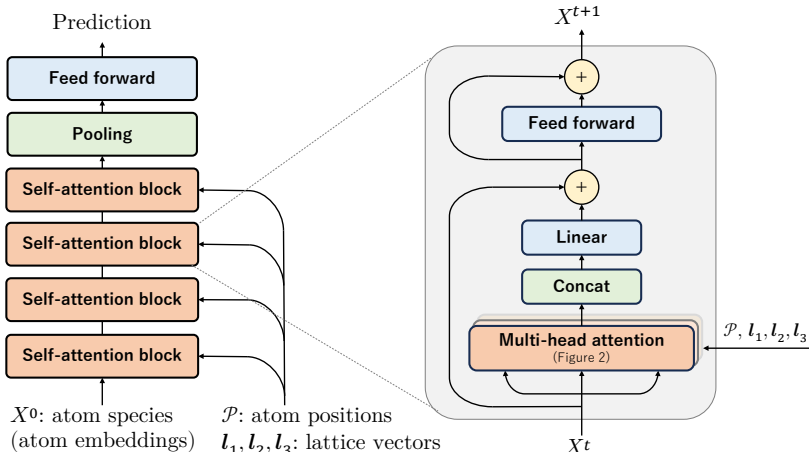


Figure 3: Network architecture of Crystalformer.

unit cells), but structures with small unit cells often require larger ranges. We thus adaptively change the range of  $n$  for each  $i$  to sufficiently cover a radius of  $3.5\sigma_i$  in Å. This is done by setting the range of  $n_1$  as  $\max(R_1, 2) \leq n_1 \leq \max(R_1, 2)$  where  $R_1 = \lceil 3.5\sigma_i k l_2 \cdot l_3 k / \det(l_1, l_2, l_3) \rceil$  ( $\lceil \cdot \rceil$  is the ceiling function) and doing similarly for  $n_2$  and  $n_3$ . For Eq. (8) we use  $K = 64$  and  $r_{\max} = 14\text{Å}$ .

### 3.2 NETWORK ARCHITECTURE

As illustrated in Fig. 3, the Crystalformer architecture basically follows the original Transformer encoder architecture (Vaswani et al., 2017) with stacked self-attention blocks, each consisting of two residual blocks connecting a multi-head attention (MHA) layer and a shallow feed-forward network (FFN). As an important difference from the original architecture, our self-attention block entirely removes Layer Normalization. We found that a normalization-free architecture with an improved weight initialization strategy proposed by Huang et al. (2020) is beneficial to stabilize the training.

Given a set of trainable embedding vectors (atom embeddings) representing the species of the unit-cell atoms as initial state  $X^0$ , Crystalformer transforms it to abstract state  $X^0$  through four stacked self-attention blocks using the attention formulation provided in Sec. 3.1. The atom-wise states in  $X^0$  are then aggregated into a single vector via the global average pooling. This material-wise feature is further converted through a FFN of linear, ReLU, and final linear layers to predict the target properties. More architectural details are provided in Appendix E.

## 4 RELATED WORK

**Invariant encoders.** Crystal structure encoders for property prediction must satisfy various invariance properties against artificial differences in data representations. The most elementary one is the permutation invariance, which was initially studied by Zaheer et al. (2017) and Qi et al. (2017) and is widely adopted into the GNN framework. Our method is permutation-invariant thanks to the Transformer architecture. The translation and rotation invariance, or the SE(3) invariance, are also essential for ML on 3D point clouds, including molecules and materials. Our method simply employs the fully distance-based modeling (Xie & Grossman, 2018) to ensure the SE(3) invariance. Recent point-cloud encoders, such as convolution-based (Zhang et al., 2019; KIM et al., 2020; Xu et al., 2021) and Transformer-based (Qin et al., 2022; Yu et al., 2023) methods, exploit richer information while maintaining the SE(3) invariance. These techniques can be possibly incorporated to our framework with proper extensions for crystal structures. Lastly, the periodic invariance (*i.e.*, supercell invariance and periodic-boundary shift invariance) has recently been pointed out by Yan et al. (2022) as a property particularly important for crystal structure encoders. Our formalism provides a periodic-invariant encoder, if  $\alpha$  and  $\beta$  are computed until convergence (see Appendix F).

**GNNs for crystal structure encoding.** The initial successful work on encoding crystal structures with neural networks can be traced back to CGCNN by Xie & Grossman (2018). Their multi-edge distance graph represents crystal structures as finite graphs, and has provided a foundation for the GNN framework for crystal structures. Almost concurrently, Schütt et al. (2018) extended their molecular encoder, SchNet, for crystal structures using a similar approach. Subsequently, several GNNs adopting similar approaches have been proposed for universal property prediction not limited to energy prediction. MEGNet (Chen et al., 2019) proposed to sequentially update atom, bond, and global state attributes through GNN layers. GATGNN (Louis et al., 2020) incorporated attention mechanisms into convolution and pooling layers of the GNN framework. iCGCNN (Park & Wolverton, 2020) and ALIGNN (Choudhary & DeCost, 2021) proposed 3-body interaction GNNs to exploit interatomic angular information, while GeoCGNN (Cheng et al., 2021) used rotation-invariant plane-wave-based edge features to encode directional information. Kosmala et al. (2023) recently proposed Ewald message passing for exploiting long-range interatomic interactions with GNNs. PotNet (Lin et al., 2023) proposed a new type of physics-informed edge feature that embeds the infinite summation value of several known interatomic potentials and allows a standard fully-connected GNN to be informed of crystal periodicity. (We more discuss PotNet in Appendix G.) Apart from material-level property prediction, M3GNet (Chen & Ong, 2022) extended MEGNet for interatomic potential prediction. Our focus is apart from the line of these studies in the GNN framework, and we aim to establish an effective Transformer framework for crystal structure encoding in a standard fashion of the Transformer using fully connected attention.

**Transformers.** The Transformer was originally proposed by Vaswani et al. (2017) for machine translation in natural language processing (NLP), as a sequence-to-sequence model in an encoder-decoder architecture. Since then, it has been widely applied to various tasks in many fields, such as NLP, computer vision, and speech processing (Lin et al., 2022), using its encoder-decoder or encoder-only architecture. Compared to standard convolutional neural network encoders, Transformer encoders have an outstanding ability to model complex interdependencies among input elements (*e.g.*, resolving ambiguous meanings of words in sentence context) and have great flexibility in handling irregularly structured or unordered data such as point clouds (Zeng et al., 2022). These capabilities should benefit the encoding of crystal structures and molecules because atoms in these structures interact with each other in complex ways to determine their states, and also they are structured in 3D space rather than regularly ordered. While there have been attempts to partly replace key modules of GNNs with attention mechanisms, Ying et al. (2021) first presented a complete Transformer architecture (Graphormer) for graph encoding and showed state-of-the-art performance for molecular property prediction. Later, Yan et al. (2022) proposed Matformer as a Transformer-inspired encoder for crystal structures. Matformer fixes the periodic invariance break in existing GNNs by using radius nearest neighbors (radius-NN) instead of k-NN for graph construction. Note that works on language models for materials by Wei et al. (2022) and Fu et al. (2023) are clearly distinguished from ours.

**Graphormer** (Ying et al., 2021) is specifically designed for finite graphs such as molecules and does not ensure distance-decay attention, which leads to its inapplicability to periodic structures. Moreover, even an extended model mending this problem suffers from another modeling problem of periodicity, such as discussed in Sec.3.1. The problem is that Graphormer encodes all the spatial information, including distance edge features, as only softmax biases (*i.e.*,  $\phi_{ij}$  in Eq. (2)) and does not use value position encoding (*i.e.*,  $\psi_{ij}$ ). Such a formulation is fine for molecules but fails to distinguish between crystal structures of the same single atom in differently sized unit cells. We will show that this modeling issue leads to performance degradation in Sec. 5.4.

**Matformer** (Yan et al., 2022) employs finite graphs with radius-NN edges similarly to GNNs. Their abstract feature-based attention does not ensure distance decay and is thus inapplicable to the infinitely connected edges. Moreover, they modify the original Transformer in various ways, by extending query/key/value by concatenation as  $\mathbf{q}_{ij} = (\mathbf{q}_i/\mathbf{q}_j/\mathbf{q}_i)$ ,  $\mathbf{k}_{ij} = (\mathbf{k}_i/\mathbf{k}_j/e_{ij})$ ,  $\mathbf{v}_{ij} = (\mathbf{v}_i/\mathbf{v}_j/e_{ij})$  ( $e_{ij}$  an edge feature), changing the scaled dot product to Hadamard product, softmax to sigmoid function, etc. We consider that the lack of explicit distance decay attention required such modifications, as they report worse results with standard softmax attention. They also propose self-connecting edges (*i.e.*, a part of the orange dashed edges in Fig. 1 (right)) although unsuccessful. By contrast, we employ much denser edges made possible by explicit distance-decay attention. This formulation leads to an architectural framework closely following the original Transformer (see Figs. 2 and 3), specifically its relative-position-based (Shaw et al., 2018) and normalization-free (Huang et al., 2020) variants.

## 5 EXPERIMENTS

We perform regression tasks of several important material properties, comparing with several neural-network-based state-of-the-art methods (Xie & Grossman, 2018; Schütt et al., 2018; Chen et al., 2019; Louis et al., 2020; Chen & Ong, 2022; Choudhary & DeCost, 2021; Yan et al., 2022; Lin et al., 2023). Following our most relevant work by Yan et al. (2022), we use the following datasets with DFT-calculated properties.

**Materials Project (MEGNet)** is a collection of 69,239 materials from the Materials Project database retrieved by Chen et al. (2019). Following Yan et al. (2022), we perform regression tasks of formation energy, bandgap, bulk modulus, and shear modulus.

**JARVIS-DFT (3D 2021)** is a collection of 55,723 materials by Choudhary et al. (2020). Following Yan et al. (2022), we perform regression tasks of formation energy, total energy, bandgap, and energy above hull (E hull). For bandgap, the dataset provides property values obtained by DFT calculation methods using the OptB88vdW functional (OPT) or the Tran-Blaha modified Becke-Johnson potential (MBJ). While MBJ is considered more accurate (Choudhary et al., 2020), we use both for evaluation.

Thanks to the great effort by Choudhary & DeCost (2021) and Yan et al. (2022), many relevant methods are evaluated on these datasets with consistent and reproducible train/validation/test splits. We partly borrow their settings and results for a fair comparison. Doing so also helps to reduce the computational load needed for comparisons.

### 5.1 TRAINING SETTINGS

For each regression task in the Materials Project dataset, we train our model by optimizing the mean absolute error loss function via stochastic gradient descent (SGD) with a batch size of 128 materials for 500 epochs. We initialize the attention layers by following Huang et al. (2020). We use the Adam optimizer (Kingma & Ba, 2015) with weight decay (Loshchilov & Hutter, 2019) of  $10^{-5}$  and clip the gradient norm at 1. For the hyperparameters of Adam we follow Huang et al. (2020); we use the initial learning rate  $\alpha$  of  $5 \times 10^{-4}$  and decay it according to  $\alpha \sqrt{4000/(4000 + t)}$  by the number of total training steps  $t$ , and use  $(\beta_1, \beta_2) = (0.9, 0.98)$ . For test and validation model selection, we use stochastic weight averaging (SWA) (Izmailov et al., 2018) by averaging the model weights for the last 50 epochs with a fixed learning rate. For the JARVIS-DFT dataset, we use increased batch size and total epochs, as specified in Appendix H, for its relatively smaller dataset size.

### 5.2 CRYSTAL PROPERTY PREDICTION

Tables 1 and 2 summarize the mean absolute errors (MAEs) for totally nine regression tasks of the Materials Project and JARVIS-DFT datasets, comparing our method with eight existing methods<sup>1</sup>. Our method consistently outperforms all but PotNet in all the tasks, even without SWA (*i.e.*, evaluating a model checkpoint with the best validation score). Meanwhile, our method is competitive to PotNet. This is rather remarkable, considering that 1) PotNet exploits several known forms of interatomic potentials to take advantage of a strong inductive bias while we approximate them with simple Gaussian potentials, and that 2) our model is more efficient as shown in the next section. It is also worth noting that our method performs well for the bulk and shear modulus prediction tasks, even with limited training data sizes without any pre-training, which supports the validity of our model.

### 5.3 MODEL EFFICIENCY COMPARISON

Table 3 summarizes the running times<sup>2</sup> and model sizes, comparing with PotNet (Lin et al., 2023) and Matformer (Yan et al., 2022) as the current best GNN-based and Transformer-based models. Notably, the total number of parameters in our model is only 48.6% and 29.4% of the number in

<sup>1</sup>Except for M3GNet, their MAEs are cited from previous works (Choudhary & DeCost, 2021; Yan et al., 2022; Lin et al., 2023). We ran M3GNet by using the code provided in Materials Graph Library (<https://github.com/materialsvirtuallab/matgl>).

<sup>2</sup>We evaluated the running times of all the methods by using a single NVIDIA RTX A6000 and a single CPU core on the same machine. We used the codes of PotNet and Matformer in the AIRS OpenMat library (Zhang et al., 2023). The inference time of PotNet is dominated by the precomputation of potential summations (309 ms).

Table 1: **MAE comparison on the Materials Project (MEGNet) dataset.** The three numbers below each property name indicate the sizes of training, validation, and testing subsets. The results of the proposed method without stochastic weight averaging (SWA) are also shown. The numbers in **bold** indicate the best results and the numbers with underline indicate the second best results.

Method	Formation energy	Bandgap	Bulk modulus	Shear modulus
	60000 / 5000 / 4239	60000 / 5000 / 4239	4664 / 393 / 393	4664 / 392 / 393
	eV/atom	eV	log(GPa)	log(GPa)
CGCNN (Xie & Grossman, 2018)	0.031	0.292	0.047	0.077
SchNet (Schütt et al., 2018)	0.033	0.345	0.066	0.099
MEGNet (Chen et al., 2019)	0.030	0.307	0.060	0.099
GATGNN (Louis et al., 2020)	0.033	0.280	0.045	0.075
M3GNet Chen & Ong (2022)	0.024	0.247	0.050	0.087
ALIGNN (Choudhary & DeCost, 2021)	0.022	0.218	0.051	0.078
Matformer (Yan et al., 2022)	0.021	0.211	0.043	0.073
PotNet (Lin et al., 2023)	<u>0.0188</u>	0.204	0.040	<b>0.065</b>
Crystalformer (proposed)	<b>0.0186</b>	<b>0.198</b>	<b>0.0377</b>	<u>0.0689</u>
Crystalformer w/o SWA	0.0198	<u>0.201</u>	<u>0.0399</u>	0.0692

Table 2: **MAE comparison on the JARVIS-DFT 3D 2021 dataset.** The bandgap prediction is evaluated with the ground truth values provided by two DFT calculation methods, OPT and MBJ.

Method	Form. energy	Total energy	Bandgap (OPT)	Bandgap (MBJ)	E hull
	44578 / 5572 / 5572	44578 / 5572 / 5572	44578 / 5572 / 5572	14537 / 1817 / 1817	44296 / 5537 / 5537
	eV/atom	eV/atom	eV	eV	eV
CGCNN	0.063	0.078	0.20	0.41	0.17
SchNet	0.045	0.047	0.19	0.43	0.14
MEGNet	0.047	0.058	0.145	0.34	0.084
GATGNN	0.047	0.056	0.17	0.51	0.12
M3GNet	0.039	0.041	0.145	0.362	0.095
ALIGNN	0.0331	0.037	0.142	0.31	0.076
Matformer	0.0325	0.035	0.137	0.30	0.064
PotNet	<b>0.0294</b>	<b>0.032</b>	<b>0.127</b>	<b>0.27</b>	0.055
Crystalformer (proposed)	0.0306	<b>0.0320</b>	0.128	0.274	<b>0.0463</b>
Crystalformer w/o SWA	0.0319	<u>0.0342</u>	0.131	0.275	<u>0.0482</u>

Table 3: **Efficiency comparison.** We show the average training time per epoch, total training time, and average testing time per material, all evaluated on the JARVIS-DFT formation energy dataset using a single NVIDIA A6000 GPU. We also show the total and number of parameters and number of parameters per self-attention or graph convolution block.

Model	Type	Time/Epoch	Total	Test/Mater.	# Params.	# Params./Block
PotNet	GNN	43 s	5.9 h	313 ms	1.8 M	527 K
Matformer	Transformer	60 s	8.3 h	20.4 ms	2.9 M	544 K
Crystalformer	Transformer	32 s	7.2 h	6.6 ms	853 K	206 K
— w/o $\psi / \beta$	Transformer	12 s	2.6 h	5.9 ms	820 K	198 K

Table 4: **Ablation studies on the validation sets of the JARVIS-DFT 3D 2021 dataset.**

Settings	$\psi$	# Blocks	Form. E.	Total E.	Bandgap (OPT)	Bandgap (MBJ)	E hull
Proposed	×	4	<b>0.0301</b>	<b>0.0314</b>	<b>0.133</b>	<b>0.287</b>	<b>0.0487</b>
Simplified		4	0.0541	0.0546	0.140	0.308	0.0517

PotNet and Matformer, respectively. We believe that our neural potential summation induces a strong inductive bias and leads to a more compact model. Although our total training time is relatively long, our inference speed is the fastest. Note that PotNet reduces its training time by precomputing infinite potential summations for training data. For reference, we also show the efficiency of our model without  $\psi/\beta$ , indicating that the computations for  $\beta$  largely dominate the entire training process.

#### 5.4 ABLATION STUDIES

We compare the proposed model with its simplified version removing value position encoding  $\psi$ , as a variant close to Graphormer (Ying et al., 2021). Table 4 shows consistent performance improvements



by the inclusion of  $\psi$ , most notably on the formation and total energy predictions but less so on the others. In Appendix I, we discuss this behavior in terms of the definitions of these metrics, and also show further performance improvements by changing the number of self-attention blocks.

## 6 LIMITATIONS AND DISCUSSION

**Angular and directional information.** We currently adopt fully distance-based formulations for position encodings  $\phi$  and  $\psi$  to ensure SE(3) invariance. Although straightforward, such formulations limit the expressive power and the addition of angular/directional information is preferred (Pozdnyakov & Ceriotti, 2022). Some existing works on SE(3)-invariant GNNs explore this direction by using 3-body interactions (Park & Wolverton, 2020; Choudhary & DeCost, 2021; Chen & Ong, 2022) or plane-wave-based edge features (Cheng et al., 2021). Others propose SE(3)-equivariant encoding techniques for a similar purpose (Liao & Smidt, 2023; Duval et al., 2023). Extending Crystalformer to a 3-body, higher-order Transformer (Kim et al., 2021), incorporating plane wave features into  $\psi$ , or introducing SE(3)-equivariant transformations are possible future directions.

**Forms of interatomic potentials.** Explicitly exposing distance-decay functions in attention helps the physical interpretation of our model. Our current setting limits them to the Gaussian decay function, intending to approximate the sum of various potentials in the real world, such as the Coulomb potential ( $1/r$ ) and the van der Waals potential ( $1/r^6$ ). Despite the simplicity, the experiments show empirically good results, and we believe that the overall Transformer architecture (*i.e.*, MHA, FFN, and repeated attention blocks) helps to learn more complex potential forms than the Gaussian function. Still, we can explore other choices of functions by explicitly incorporating known potential forms into our model. This can be done by assigning different forms of  $\exp(\phi)$  for different heads of MHA, which possibly leads to further performance gains or model efficiency. We leave it as our future work.

**Attention in Fourier space for long-range interactions.** Gaussian tail length  $\sigma_i$  in Eq. (7) is upper-bounded by a certain constant,  $\sigma_{\text{ub}} \approx 2\text{\AA}$ , to avoid excessively long-tailed functions, thus making  $\alpha$  and  $\beta$  computationally tractable. Despite the empirical success, such a bound possibly overlooks the presence of long-range interactions, such as the Coulomb potential. Here, our infinite-series formulation becomes advantageous for computing such long-range interactions, with the help of *reciprocal space*. Reciprocal space, or Fourier space, is analogous to the frequency domain of time-dependent functions, and appears in the 3D Fourier transform of spatial functions. When Eq. (5) is written as  $\alpha_{ij} = \log f(\mathbf{p}_j - \mathbf{p}_i)$  with spatial function  $f(\mathbf{r})$ ,  $f$  is a periodic function that can be expressed in reciprocal space via Fourier series. In reciprocal space,  $f$ 's infinite series of Gaussian functions of distances in real space becomes an infinite series of Gaussian functions of spatial frequencies. (See Appendix J for detailed formulations.) These two expressions are complementary in that short-tail and long-tail potentials decay rapidly in real and reciprocal space, respectively. We use these two expressions for parallel heads of each MHA, by computing their  $\alpha$  and  $\beta$  differently in real or reciprocal space. To ensure the tractability in reciprocal space,  $\sigma_i$  should be lower-bounded as  $\sigma_i > \sigma_{\text{lb}}$ . By setting the bounds for the two spaces as  $\sigma_{\text{lb}} < \sigma_{\text{ub}}$ , this dual-space MHA can cover the entire range of interactions in theory. As a tentative experiment, we evaluate this dual-space variant of Crystalformer on the JARVIS-DFT dataset. Results in Table A6 in Appendix J.4 show that adding the reciprocal-space attention results in worse performance for formation and total energy, comparable for bandgap, and much better for E hull predictions. Adaptively switching each MHA head between the two attention forms depending on the task will lead to stable improvements.

## 7 CONCLUSIONS

We have presented Crystalformer as a Transformer encoder for crystal structures. It stands on fully connected attention between periodic points, namely infinitely connected attention, with physically-motivated distance-decay attention to ensure the tractability. The method has shown successful results in various property prediction tasks with high model efficiency. We have further discussed its extension to a reciprocal space representation for efficiently computing long-range interatomic interactions. We hope that this simple and physically-motivated Transformer framework provides a perspective in terms of both ML and materials science to promote further interdisciplinary research. As Transformer-based large language models are revolutionizing AI, how Crystalformer with large models can absorb knowledge from large-scale material datasets is also an ambitious open question.

## AUTHOR CONTRIBUTIONS

T.T. conceived the core idea of the method, implemented it, conducted the experiments, and wrote the paper and rebuttal with other authors. R.I. suggested the Fourier space attention, provided knowledge of physics simulation and materials science, and helped with the paper writing and rebuttal. Y.S. provided a codebase for crystal encoders, provided materials science knowledge, and helped with the experiments, survey of crystal encoders, and paper writing. N.C. provided ML knowledge and helped with the experiments, survey of invariant encoders, and paper writing. K.S. provided materials science knowledge and helped with the paper writing. Y.U. co-led our materials-related collaborative projects, provided ML knowledge, and helped with the experiments and paper writing. K.O. co-led our materials-related collaborative projects, provided materials science knowledge, and helped with the paper writing.

## ACKNOWLEDGMENTS

This work is partly supported by JST-Mirai Program, Grant Number JPMJMI19G1. Y.S. was supported by JST ACT-I grant number JPMJPR18UE during the early phase of this project. The computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by the National Institute of Advanced Industrial Science and Technology (AIST) was partly used for numerical experiments.

## REFERENCES

- Chi Chen and Shyue Ping Ong. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science*, 2(11):718–728, Nov 2022. ISSN 2662-8457. doi: 10.1038/s43588-022-00349-3. URL <https://doi.org/10.1038/s43588-022-00349-3>.
- Chi Chen, Weike Ye, Yunxing Zuo, Chen Zheng, and Shyue Ping Ong. Graph networks as a universal machine learning framework for molecules and crystals. *Chemistry of Materials*, 31(9):3564–3572, May 2019. ISSN 0897-4756. doi: 10.1021/acs.chemmater.9b01294.
- Jiucheng Cheng, Chunkai Zhang, and Lifeng Dong. A geometric-information-enhanced crystal graph network for predicting properties of materials. *Communications Materials*, 2(1):92, Sep 2021. ISSN 2662-4443. doi: 10.1038/s43246-021-00194-3. URL <https://doi.org/10.1038/s43246-021-00194-3>.
- Kamal Choudhary and Brian DeCost. Atomistic line graph neural network for improved materials property predictions. *npj Computational Materials*, 7(1):185, Nov 2021. ISSN 2057-3960. doi: 10.1038/s41524-021-00650-1. URL <https://doi.org/10.1038/s41524-021-00650-1>.
- Kamal Choudhary, Kevin F. Garrity, Andrew C. E. Reid, Brian DeCost, Adam J. Biacchi, Angela R. Hight Walker, Zachary Trautt, Jason Hattrick-Simpers, A. Gilad Kusne, Andrea Centrone, Albert Davydov, Jie Jiang, Ruth Pachter, Gowoon Cheon, Evan Reed, Ankit Agrawal, Xiaofeng Qian, Vinit Sharma, Houlong Zhuang, Sergei V. Kalinin, Bobby G. Sumpter, Ghanshyam Pilania, Pinar Acar, Subhasish Mandal, Kristjan Haule, David Vanderbilt, Karin Rabe, and Francesca Tavazza. The joint automated repository for various integrated simulations (jarvis) for data-driven materials design. *npj Computational Materials*, 6(1):173, Nov 2020. ISSN 2057-3960. doi: 10.1038/s41524-020-00440-1. URL <https://doi.org/10.1038/s41524-020-00440-1>.
- Kamal Choudhary, Brian DeCost, Chi Chen, Anubhav Jain, Francesca Tavazza, Ryan Cohn, Cheol Woo Park, Alok Choudhary, Ankit Agrawal, Simon J. L. Billinge, Elizabeth Holm, Shyue Ping Ong, and Chris Wolverton. Recent advances and applications of deep learning methods in materials science. *npj Computational Materials*, 8(1):59, Apr 2022. ISSN 2057-3960. doi: 10.1038/s41524-022-00734-6. URL <https://doi.org/10.1038/s41524-022-00734-6>.
- Bernard Deconinck, Matthias Heil, Alexander Bobenko, Mark van Hoeij, and Marcus Schmies. Computing riemann theta functions. *Mathematics of Computation*, 73(247):1417–1442, 2004. ISSN 00255718, 10886842. URL <http://www.jstor.org/stable/4099903>.
- Claudia Draxl and Matthias Scheffler. The nomad laboratory: from data sharing to artificial intelligence. *Journal of Physics: Materials*, 2(3):036001, may 2019. doi: 10.1088/2515-7639/ab13bb. URL <https://dx.doi.org/10.1088/2515-7639/ab13bb>.

- Alexandre Agm Duval, Victor Schmidt, Alex Hernández-García, Santiago Miret, Fragkiskos D. Malliaros, Yoshua Bengio, and David Rolnick. FAENet: Frame averaging equivariant GNN for materials modeling. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 9013–9033. PMLR, 23–29 Jul 2023. URL <https://proceedings.mlr.press/v202/duval23a.html>.
- Nihang Fu, Lai Wei, Yuqi Song, Qinyang Li, Rui Xin, Sadman Sadeed Omea, Rongzhi Dong, Edirisuriya M. Dilanga Siriwardane, and Jianjun Hu. Material transformers: deep learning language models for generative materials design. *Mach. Learn. Sci. Technol.*, 4(1):15001, 2023. doi: 10.1088/2632-2153/acadcd. URL <https://doi.org/10.1088/2632-2153/acadcd>.
- Xiao Shi Huang, Felipe Perez, Jimmy Ba, and Maksims Volkovs. Improving transformer optimization through better initialization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4475–4483. PMLR, 13–18 Jul 2020.
- Pavel Izmailov, Dmitrii Podoprikin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization, 2018. URL <http://arxiv.org/abs/1803.05407>.
- Jinwoo Kim, Saeyoon Oh, and Seunghoon Hong. Transformers generalize deepsets and can be extended to graphs & hypergraphs. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 28016–28028. Curran Associates, Inc., 2021. URL [https://proceedings.neurips.cc/paper\\_files/paper/2021/file/ec0f40c389aeeef789ce03eb814facc6c-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2021/file/ec0f40c389aeeef789ce03eb814facc6c-Paper.pdf).
- SEOHYUN KIM, JaeYoo Park, and Bohyung Han. Rotation-invariant local-to-global representation learning for 3d point cloud. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 8174–8185. Curran Associates, Inc., 2020. URL [https://proceedings.neurips.cc/paper\\_files/paper/2020/file/5d0cb12f8c9ad6845110317afc6e2183-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2020/file/5d0cb12f8c9ad6845110317afc6e2183-Paper.pdf).
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL <http://arxiv.org/abs/1412.6980>.
- Scott Kirklin, James E. Saal, Bryce Meredig, Alex Thompson, Jeff W. Doak, Muratahan Aykol, Stephan Rühl, and Chris Wolverton. The open quantum materials database (oqmd): assessing the accuracy of dft formation energies. *npj Computational Materials*, 1(1):15010, Dec 2015. ISSN 2057-3960. doi: 10.1038/npjcompumats.2015.10. URL <https://doi.org/10.1038/npjcompumats.2015.10>.
- Arthur Kosmala, Johann Gasteiger, Nicholas Gao, and Stephan Gunnemann. Ewald-based long-range message passing for molecular graphs. In *International Conference on Machine Learning*, 2023.
- Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=KwmPfARgOTD>.
- Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. A survey of transformers. *AI Open*, 3: 111–132, 2022. ISSN 2666-6510. doi: <https://doi.org/10.1016/j.aiopen.2022.10.001>. URL <https://www.sciencedirect.com/science/article/pii/S2666651022000146>.
- Yuchao Lin, Keqiang Yan, Youzhi Luo, Yi Liu, Xiaoning Qian, and Shuiwang Ji. Efficient approximations of complete interatomic potentials for crystal property prediction. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21260–21287. PMLR, 23–29 Jul 2023. URL <https://proceedings.mlr.press/v202/lin23m.html>.

- Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL <https://openreview.net/forum?id=Bkg6RiCqY7>.
- Steph-Yves Louis, Yong Zhao, Alireza Nasiri, Xiran Wang, Yuqi Song, Fei Liu, and Jianjun Hu. Graph convolutional neural networks with global attention for improved materials property prediction. *Phys. Chem. Chem. Phys.*, 22:18141–18148, 2020. doi: 10.1039/D0CP01474E. URL <http://dx.doi.org/10.1039/D0CP01474E>.
- Cheol Woo Park and Chris Wolverton. Developing an improved crystal graph convolutional neural network framework for accelerated materials discovery. *Phys. Rev. Mater.*, 4:063801, Jun 2020. doi: 10.1103/PhysRevMaterials.4.063801. URL <https://link.aps.org/doi/10.1103/PhysRevMaterials.4.063801>.
- Robert Pollice, Gabriel dos Passos Gomes, Matteo Aldeghi, Riley J. Hickman, Mario Krenn, Cyrille Lavigne, Michael Lindner-D’Addario, AkshatKumar Nigam, Cher Tian Ser, Zhenpeng Yao, and Alán Aspuru-Guzik. Data-driven strategies for accelerated materials design. *Accounts of Chemical Research*, 54(4):849–860, Feb 2021. ISSN 0001-4842. doi: 10.1021/acs.accounts.0c00785. URL <https://doi.org/10.1021/acs.accounts.0c00785>.
- Sergey N Pozdnyakov and Michele Ceriotti. Incompleteness of graph neural networks for points clouds in three dimensions. *Machine Learning: Science and Technology*, 3(4):045020, nov 2022. doi: 10.1088/2632-2153/acaf8. URL <https://dx.doi.org/10.1088/2632-2153/acaf8>.
- Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 77–85. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.16. URL <https://doi.org/10.1109/CVPR.2017.16>.
- Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- Patrick Reiser, Marlen Neubert, André Eberhard, Luca Torresi, Chen Zhou, Chen Shao, Houssam Metni, Clint van Hoesel, Henrik Schopmans, Timo Sommer, and Pascal Friederich. Graph neural networks for materials science and chemistry. *Communications Materials*, 3(1):93, Nov 2022. ISSN 2662-4443. doi: 10.1038/s43246-022-00315-6. URL <https://doi.org/10.1038/s43246-022-00315-6>.
- K. T. Schütt, H. E. Saucedo, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, 03 2018. ISSN 0021-9606. doi: 10.1063/1.5019779. URL <https://doi.org/10.1063/1.5019779>.
- Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Saucedo Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.
- Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *North American Chapter of the Association for Computational Linguistics*, 2018. URL <https://api.semanticscholar.org/CorpusID:3725815>.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf).

- Lai Wei, Qinyang Li, Yuqi Song, Stanislav Stefanov, Edirisuriya M. Dilanga Siriwardane, Fanglin Chen, and Jianjun Hu. Crystal transformer: Self-learning neural language model for generative and tinkering design of materials. *CoRR*, abs/2204.11953, 2022. doi: 10.48550/arXiv.2204.11953. URL <https://doi.org/10.48550/arXiv.2204.11953>.
- Tian Xie and Jeffrey C. Grossman. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys. Rev. Lett.*, 120:145301, Apr 2018. doi: 10.1103/PhysRevLett.120.145301. URL <https://link.aps.org/doi/10.1103/PhysRevLett.120.145301>.
- Jianyun Xu, Xin Tang, Yushi Zhu, Jie Sun, and Shiliang Pu. Sgmnet: Learning rotation-invariant point cloud representations via sorted gram matrix. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.
- Keqiang Yan, Yi Liu, Yuchao Lin, and Shuiwang Ji. Periodic graph transformers for crystal material property prediction. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 15066–15080. Curran Associates, Inc., 2022.
- Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 28877–28888. Curran Associates, Inc., 2021. URL [https://proceedings.neurips.cc/paper\\_files/paper/2021/file/f1c1592588411002af340cbaedd6fc33-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2021/file/f1c1592588411002af340cbaedd6fc33-Paper.pdf).
- Hao Yu, Zheng Qin, Ji Hou, Mahdi Saleh, Dongsheng Li, Benjamin Busam, and Slobodan Ilic. Rotation-invariant transformer for point cloud matching. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/f22e4747dalaa27e363d86d40ff442fe-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/f22e4747dalaa27e363d86d40ff442fe-Paper.pdf).
- Jiahao Zeng, Decheng Wang, and Peng Chen. A survey on transformers for point cloud processing: An updated overview. *IEEE Access*, 10:86510–86527, 2022. doi: 10.1109/ACCESS.2022.3198999.
- Xuan Zhang, Limei Wang, Jacob Helwig, Youzhi Luo, Cong Fu, Yaochen Xie, Meng Liu, Yuchao Lin, Zhao Xu, Keqiang Yan, Keir Adams, Maurice Weiler, Xiner Li, Tianfan Fu, Yucheng Wang, Haiyang Yu, YuQing Xie, Xiang Fu, Alex Strasser, Shenglong Xu, Yi Liu, Yuanqi Du, Alexandra Saxton, Hongyi Ling, Hannah Lawrence, Hannes Stärk, Shurui Gui, Carl Edwards, Nicholas Gao, Adriana Ladera, Tailin Wu, Elyssa F. Hofgard, Aria Mansouri Tehrani, Rui Wang, Ameya Daigavane, Montgomery Bohde, Jerry Kurtin, Qian Huang, Tuong Phung, Minkai Xu, Chaitanya K. Joshi, Simon V. Mathis, Kamyar Azizzadenesheli, Ada Fang, Alán Aspuru-Guzik, Erik Bekkers, Michael Bronstein, Marinka Zitnik, Anima Anandkumar, Stefano Ermon, Pietro Liò, Rose Yu, Stephan Gunnemann, Jure Leskovec, Heng Ji, Jimeng Sun, Regina Barzilay, Tommi Jaakkola, Connor W. Coley, Xiaoning Qian, Xiaofeng Qian, Tess Smidt, and Shuiwang Ji. Artificial intelligence for science in quantum, atomistic, and continuum systems. *arXiv preprint arXiv:2307.08423*, 2023. URL <https://github.com/divelab/AIRS>.
- Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

## A DERIVATION OF PSEUDO-FINITE PERIODIC ATTENTION

Starting from Eq. (3), we can show that

$$\mathbf{y}_i = \frac{1}{Z_i} \sum_{j=1}^N \sum_{\mathbf{n}} \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \phi_{ij}(\mathbf{n})) (\mathbf{v}_{j(\mathbf{n})} + \boldsymbol{\psi}_{ij(\mathbf{n})}) \quad (\text{A1})$$

$$= \frac{1}{Z_i} \sum_{j=1}^N \sum_{\mathbf{n}} \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \phi_{ij}(\mathbf{n})) (\mathbf{v}_j + \boldsymbol{\psi}_{ij(\mathbf{n})}) \quad (\text{A2})$$

$$= \frac{1}{Z_i} \sum_{j=1}^N \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K}) \sum_{\mathbf{n}} \exp(\phi_{ij}(\mathbf{n})) (\mathbf{v}_j + \boldsymbol{\psi}_{ij(\mathbf{n})}) \quad (\text{A3})$$

$$= \frac{1}{Z_i} \sum_{j=1}^N \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K}) \exp(\alpha_{ij}) (\mathbf{v}_j + \frac{\sum_{\mathbf{n}} \exp(\phi_{ij}(\mathbf{n})) \boldsymbol{\psi}_{ij(\mathbf{n})}}{\exp(\alpha_{ij})}) \quad (\text{A4})$$

$$= \frac{1}{Z_i} \sum_{j=1}^N \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \alpha_{ij}) (\mathbf{v}_j + \boldsymbol{\beta}_{ij}), \quad (\text{A5})$$

which equals Eq. (4).

## B BOUND FOR APPROXIMATION ERROR

The computations of  $\alpha_{ij}$  and  $\boldsymbol{\beta}_{ij}$  involve the following infinite summation:

$$Z_{ij} = \sum_{\mathbf{n}} \exp\left(\frac{k\mathbf{p}_j + L\mathbf{n} \cdot \mathbf{p}_i k^2}{2\sigma_i^2}\right) \quad (\text{A6})$$

where  $L = [\ell_1, \ell_2, \ell_3]$ . As explained in the implementation details (Sec. 3.1), we approximate  $Z_{ij}$  by finite summation:

$$\tilde{Z}_{ij} = \sum_{n_1=1}^{R_1} \sum_{n_2=1}^{R_2} \sum_{n_3=1}^{R_3} \exp\left(\frac{k\mathbf{p}_j + L\mathbf{n} \cdot \mathbf{p}_i k^2}{2\sigma_i^2}\right), \quad (\text{A7})$$

where ranges  $R_1, R_2, R_3 \geq 2$  are adaptively determined so that the summed points (*i.e.*,  $\mathbf{p}_j + L\mathbf{n} \cdot \mathbf{p}_i$ ) sufficiently cover a spherical radius of  $c\sigma_i$  in 3D space ( $c$  is set to 3.5) while ensuring  $R_1, R_2, R_3 \geq 2$ . This derives an upper bound for the residual error between  $Z_{ij}$  and  $\tilde{Z}_{ij}$  as

$$\epsilon_{ij} = Z_{ij} - \tilde{Z}_{ij} = \sum_{r \geq 2\Lambda_{ij}, krk \geq R} \exp(-krk^2), \quad (\text{A8})$$

where  $\Lambda_{ij} = \frac{1}{2\sigma_i} (L\mathbf{n} + \mathbf{p}_j - \mathbf{p}_i) \cdot \mathbf{n} \geq Z^3 g$ ,  $R = \max(c/\sqrt{2}, R^0)$ ,  $R^0 = \min\left(\frac{1}{2\sigma_i} k\mathbf{p}_j + L\mathbf{n} \cdot \mathbf{p}_i k \cdot \mathbf{n} \geq Z^3, \max(j_{n_1}, j_{n_2}, j_{n_3})\right) > 2g$ .

Here, Deconinck et al. (2004) prove the following inequality:

$$\sum_{r \geq 2\Lambda, krk \geq R} krk^p \exp(-krk^2) \leq \frac{d+p}{2} \left(\frac{2}{\rho}\right)^d \left(\frac{d}{2}, \left(R - \frac{\rho}{2}\right)^2\right), \quad (\text{A9})$$

where  $\rho = \sqrt{X\mathbf{n} + \mathbf{x}} \cdot \mathbf{n} \geq Z^d g$ ,  $X \geq R^d$  is a full-rank matrix,  $\mathbf{x} \geq R^d$ ,  $\rho = \min\sqrt{X\mathbf{n} + \mathbf{x}} \cdot \mathbf{n} \geq Z^d$ ,  $\mathbf{n} \notin \mathbf{0}g$ , and  $\Gamma(z, x) = \int_x^\infty t^{z-1} e^{-t} dt$ , the incomplete Gamma function.

Substituting  $p = 0$ ,  $d = 3$ , and  $X = \frac{1}{2\sigma_i} L$  into this inequality derives a bound for our residual error:

$$\epsilon_{ij} \leq \frac{3}{2} \left(\frac{2}{\rho}\right)^3 \left(\frac{3}{2}, \left(R - \frac{\rho}{2}\right)^2\right), \quad (\text{A10})$$

where  $\rho = \min\sqrt{\frac{1}{2\sigma_i} kL\mathbf{n}k \cdot \mathbf{n} \geq Z^d}$ ,  $\mathbf{n} \notin \mathbf{0}g$ .

## C DEFINITION OF $\sigma_i$ .

To increase the expressive power of the model, we treat tail length  $\sigma_i$  of the Gaussian potential in Eq. (7) as a variable parameterized via current state  $\mathbf{q}_i$ . Here, the exponent of the Gaussian function,  $\phi_{ij(n)} = \mathbf{k}\mathbf{p}_{j(n)} - \mathbf{p}_i\mathbf{k}^2/2\sigma_i^2$ , is intended to represent the relative position encoding by Shaw et al. (2018), whose original form is  $\phi_{ij} = \mathbf{q}_i^T \mathbf{a}_{ij}$  with embedding  $\mathbf{a}_{ij}$  of relative position  $j - i$ . As such a linear form of  $\mathbf{q}_i$  should ease the training of the network, we model  $\sigma_i^{-2}$  instead of  $\sigma_i$  as a function:

$$\sigma_i^{-2} = r_0^{-2} \rho((\mathbf{q}_i^T \mathbf{w} - m)/s) \quad (\text{A11})$$

where  $r_0$  is a hyperparameter determining a distance scale of spatial dependencies in Å,  $\mathbf{w}$  is a trainable weight vector,  $m$  and  $s$  are constant parameters, and  $\rho : \mathbb{R} \rightarrow \mathbb{R}^+$  is a function to ensure a positive value for  $\sigma_i^{-2}$ .

The adjustment of  $r_0$  could be difficult, since it has to bridge between network’s internal states (*i.e.*,  $\mathbf{q}_i^T \mathbf{w}$ ) and physical distances (*i.e.*,  $\sigma_i$ ) across different scales. To ease it, we design  $\rho$  so that the initial values of  $\rho$  and hence  $\sigma_i$  are distributed around 1 and  $r_0$ , respectively. To this end, we initialize  $m$  and  $s$  as the mean and standard deviation of the values of  $\mathbf{q}_i^T \mathbf{w}$  computed for the mini-batch samples at the initial training step. Then, for  $\rho$  we introduce a *shifted exponential linear unit* (shifted ELU):

$$\rho(x; a, b) = (1 - b)\text{ELU}(ax/(1 - b)) + 1, \quad (\text{A12})$$

where  $\text{ELU}(x) = \exp(x) - 1$  if  $x < 0$  and  $\text{ELU}(x) = x$  otherwise. Assuming that  $x$  initially follows  $\mathcal{N}(0, 1)$ , shifted ELU  $\rho(x; a, b)$  is designed to be 1)  $\rho(0) = 1$  so that  $\sigma_i$  is initially distributed around  $r_0$ , 2) lower-bounded as  $\rho(x) > b$  so that  $\sigma_i < r_0/\sqrt{b}$  to avoid excessively long tails, and 3) linear response with derivative  $a$  for  $x \geq 0$  to increase the training stability. We empirically set the hyperparameters to  $(r_0, a, b) = (1.4, 0.1, 0.5)$ .

## D NECESSITY OF VALUE POSITION ENCODING

Shaw et al. (2018) discuss that value position encoding is required depending on the application, and we show that crystal structure encoding is indeed such a case.

As mentioned in the main texts, Eq. (3) without value position encoding  $\psi_{ij(n)}$  cannot distinguish between crystal structures of the same single atom in differently sized unit cells. This can be confirmed by substituting  $X = \mathbf{f}\mathbf{x}_1\mathbf{g}$  or  $N = 1$  into Eq. (4) as below.

$$\mathbf{y}_1 = \frac{1}{Z_1} \exp(\mathbf{q}_1^T \mathbf{k}_1 / \sqrt{d_K} + \alpha_{11})(\mathbf{v}_1 + \beta_{11}) \quad (\text{A13})$$

$$= \mathbf{v}_1 + \beta_{11} \quad (\text{A14})$$

$$= \mathbf{v}_1 + \frac{\sum_{\mathbf{n}} \exp(\phi_{11(n)}) \psi_{11(n)}}{\sum_{\mathbf{n}} \exp(\phi_{11(n)})} \quad (\text{A15})$$

Note that the lattice information (*i.e.*,  $\ell_1, \ell_2, \ell_3$ ) is used only in key position encoding  $\phi_{11(n)}$  and value position encoding  $\psi_{11(n)}$  for  $\alpha_{11}$  and  $\beta_{11}$ , and does not exist in  $\mathbf{q}, \mathbf{k}, \mathbf{v}$  in the initial state. The above equations show that when  $N = 1$ , the weight of  $(\mathbf{v}_1 + \beta_{11})$  is canceled out by  $Z_1$  and therefore the lattice information is only present in  $\beta_{11}$ . Without value position encoding (*i.e.*,  $\psi_{11(n)} = \mathbf{0}$ ), the attention formula becomes  $\mathbf{y}_1 = \mathbf{v}_1$ , showing that result  $\mathbf{y}_1$  does not reflect how the atom is repeated in 3D space by  $\ell_1, \ell_2, \ell_3$ .

From a materials science perspective, crystal structures of single unit-cell atoms are called *monatomic structures* and represent an important class of existing materials. Well-known monatomic structures of the same atom include graphite and diamond. They have the single unit-cell atom of Carbon but have different lattice vectors and thus different crystal structures. These differences make graphite a soft substance used for pencil tops while making diamond the hardest substance in the world. The value position encoding is thus required to distinguish such important materials.

## E DETAILED NETWORK ARCHITECTURE

Table A1 shows the detailed specifications of the proposed Crystalformer architecture. Note that changing the ReLU activation function to GELU in the FFNs of the self-attention blocks did not lead to meaningful performance improvements.

Table A1: Detailed network architecture.

Blocks	Output dims	Specifications
Atom embedding	$(128, N)$	$\{e_k^{\text{atom}}   k = \text{AtomicNumber}(i), i = 1, 2, \dots, N\}$ where $e^{\text{atom}}$ 's are initialized by $\mathcal{N}(0, 128^{-0.5})$
Self-attention block $\times 4$	$(128, N)$	Model dim $d$ : 128 Number of heads: 8 Query/Key/Value dims: 16 FFN: Linear-ReLU-Linear (dims: $128 \rightarrow 512 \rightarrow 128$ ) Activation func: ReLU Normalization: No
Pooling	128	Global average pooling
Feed forward	1	Linear with output dim: 128 ReLU Linear with output dim: 1 (target property dim.)

## F INVARIANCE PROPERTIES

### F.1 DEFINITIONS

Following notation by Yan et al. (2022), we represent a crystal structure with a tuple of three matrices,  $(X, P, L)$ , where  $X = [\mathbf{x}_1, \dots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$  is the states of  $N$  unit-cell atoms,  $P = [\mathbf{p}_1, \dots, \mathbf{p}_N] \in \mathbb{R}^{3 \times N}$  is the 3D Cartesian coordinates of these atoms, and  $L = [\ell_1, \ell_2, \ell_3] \in \mathbb{R}^{3 \times 3}$  is the lattice vector matrix. The coordinates of the  $N$  unit-cell points are defined to be within the unit cell region defined by  $L$ , that is, they have fractional coordinates of  $L^{-1}P \in [0, 1)^3$ . When the overall network architecture is seen as function  $f(X, P, L) \rightarrow X$ , they satisfy the following invariance properties.

**Definition 1: Unit cell permutation invariance.** A function  $f : (X, P, L) \rightarrow X$  is unit cell permutation invariant such that for any permutation function  $\sigma : \{1, 2, \dots, N\} \rightarrow \{1, 2, \dots, N\}$ , we have  $f(X, P, L) = f(\sigma(X), \sigma(P), L)$ , where  $\sigma(X) = [\mathbf{x}_{\sigma(1)}, \dots, \mathbf{x}_{\sigma(N)}]$  and  $\sigma(P) = [\mathbf{p}_{\sigma(1)}, \dots, \mathbf{p}_{\sigma(N)}]$ .

**Definition 2: Unit cell E(3) invariance.** A function  $f : (X, P, L) \rightarrow X$  is unit cell E(3) invariant such that for all  $R \in \mathbb{R}^{3 \times 3}$ ,  $R^T = R^{-1}$ ,  $\det(R) = 1$ , and  $\mathbf{b} \in \mathbb{R}^3$ , we have  $f(X, P, L) = f(X, TP, RL)$ , where  $T = [R, \mathbf{b}] \in \mathbb{R}^{3 \times 4}$  is a rigid transformation,  $R$  is a rotation matrix,  $\mathbf{b}$  is a translation vector, and  $P \in \mathbb{R}^{3 \times N}$  is the homogeneous coordinates of  $P$  in 3D space. (Note that the definition gives unit cell SE(3) invariance when  $R$  is restricted to rotation matrix, as  $\det(R) = 1$ .)

Due to the arbitrary choice of extracting a unit cell structure from the infinitely expanding crystal structure in 3D space, there is an infinite number of invariants  $(\hat{X}, \hat{P}, \hat{L})$  that represent the same crystal structure as  $(X, P, L)$ . Similarly to the notation by Yan et al. (2022), we define such a unit-cell slice of a given crystal structure as

$$(\hat{X}, \hat{P}, \hat{L}) = (X, P, L, \mathbf{k}, \mathbf{p}). \quad (\text{A16})$$

The slice is parameterized via a periodic boundary scale,  $\mathbf{k} \in \mathbb{Z}_+^3$ , that defines a supercell of  $L$  as

$$\hat{L} = [k_1 \ell_1, k_2 \ell_2, k_3 \ell_3], \quad (\text{A17})$$

and a periodic boundary shift,  $\mathbf{p} \in \mathbb{R}^3$ , that provides a corner point of the periodic boundaries. The unit-cell atoms in the slice are then provided as a collection of the atoms contained in new unit cell  $\hat{L}$  with the scaled and shifted periodic boundaries as

$$\hat{P} = \hat{r} \mathbf{p}_i + L \mathbf{n} - \mathbf{p}_j \hat{L}^{-1} (\mathbf{p}_i + L \mathbf{n} - \mathbf{p}) \in [0, 1)^3, i \in \{1, \dots, N\}, \mathbf{n} \in \mathbb{Z}^3, \quad (\text{A18})$$

$$\hat{X} = \hat{r} \mathbf{x}_i \hat{L}^{-1} (\mathbf{p}_i + L \mathbf{n} - \mathbf{p}) \in [0, 1)^3, i \in \{1, \dots, N\}, \mathbf{n} \in \mathbb{Z}^3. \quad (\text{A19})$$

The periodic invariance described by Yan et al. (2022) is then summarized as follow.



**Definition 3: Periodic invariance.** A unit cell  $E(3)$  invariant function  $f : (X, P, L) \rightarrow \mathbb{R}$  is periodic invariant if  $f(X, P, L) = f(X, P, L, \mathbf{k}, \mathbf{p})$  holds for all  $\mathbf{k} \in \mathbb{Z}_+^3$  and  $\mathbf{p} \in \mathbb{R}^3$ .

## F.2 PROOFS OF INVARIANCE

**Proof of unit cell permutation invariance.** Let us represent overall network transformation  $f$  as

$$f(X, P, L) = o(h(g(X, P, L))), \quad (\text{A20})$$

where  $X^\theta = g(X, P, L)$  is the transformation by stacked self-attention blocks,  $\mathbf{z} = h(X^\theta)$  is a global pooling operation, and  $o(\mathbf{z})$  is the final output network given global feature vector  $\mathbf{z}$ . We can easily show that  $g$  is permutation equivariant such that  $\sigma(X^\theta) = g(\sigma(X), \sigma(P), L)$  holds for any permutation  $\sigma$ . (This is obvious from that fact that substituting  $i \rightarrow \sigma(i)$  and  $j(\mathbf{n}) \rightarrow \sigma(j)(\mathbf{n})$  into the right-hand side of Eq. (3) yields  $y_{\sigma(i)}$  and thus the attention operation by Eq. (3) is permutation equivariant.) We can then prove that  $f(\sigma(X), \sigma(P), L) = o(h(\sigma(X^\theta))) = o(h(X)) = f(X, P, L)$  holds if  $h$  is a permutation invariant pooling operation such as the global mean or max pooling.

**Proof of unit cell  $E(3)$  invariance.** In  $f$ , the spatial information (*i.e.*, unit-cell atom positions  $P$  and lattice vectors  $L$ ) is only used by the position encodings,  $\phi_{ij(\mathbf{n})}$  and  $\psi_{ij(\mathbf{n})}$ , in Eq. (3). Currently, they are defined to simply encode relative distances as

$$\phi_{ij(\mathbf{n})} = \phi(k\mathbf{p}_i - \mathbf{p}_{j(\mathbf{n})}), \quad (\text{A21})$$

$$\psi_{ij(\mathbf{n})} = \psi(k\mathbf{p}_i - \mathbf{p}_{j(\mathbf{n})}), \quad (\text{A22})$$

thus invariant to rigid transformation  $(P^\theta, L^\theta) = (TP, RL)$  as

$$k\mathbf{p}_i^\theta - \mathbf{p}_{j(\mathbf{n})}^\theta = k\mathbf{p}_i - (\mathbf{p}_j + L\mathbf{n}) \quad (\text{A23})$$

$$= k(R\mathbf{p}_i + \mathbf{b}) - (R\mathbf{p}_j + \mathbf{b} + RL\mathbf{n}) \quad (\text{A24})$$

$$= kR(\mathbf{p}_i - \mathbf{p}_{j(\mathbf{n})}) \quad (\text{A25})$$

$$= k\mathbf{p}_i - \mathbf{p}_{j(\mathbf{n})}. \quad (\text{A26})$$

Note that our framework allows the use of directional information by using relative position  $(\mathbf{p}_i - \mathbf{p}_{j(\mathbf{n})})$  instead of relative distance  $k\mathbf{p}_i - \mathbf{p}_{j(\mathbf{n})}$ , provided that  $\phi(\mathbf{r})$  and  $\psi(\mathbf{r})$  are rotation invariant.

**Proof of periodic invariance.** It is sufficient to show that Eq. (3) is invariant for minimum repeatable unit-cell representation  $(X, P, L)$  and its invariant representation  $(\hat{X}, \hat{P}, \hat{L}) = (X, P, L, \mathbf{k}, \mathbf{p})$ . Periodic invariance of Eq. (3) is intuitively understandable, because Eq. (3) recovers the original, infinitely expanding crystal structure from its unit cell representation and attends to all the atoms in it with relative position encoding. A formal proof is provided below.

Comparing  $(X, P, L)$  and  $(\hat{X}, \hat{P}, \hat{L})$ , each unit-cell point  $i \in \{1, \dots, Ng\}$  in  $(X, P, L)$  is populated to  $k_1 k_2 k_3$  unit-cell points in  $(\hat{X}, \hat{P}, \hat{L})$ , which we index by  $i[\mathbf{m}]$  where  $\mathbf{m} = (m_1, m_2, m_3)^T \in \mathbb{Z}^3$  and  $0 \leq m_1 \leq k_1 - 1, 0 \leq m_2 \leq k_2 - 1, 0 \leq m_3 \leq k_3 - 1$  (or simply  $\mathbf{0} \leq \mathbf{m} \leq \mathbf{k} - \mathbf{1}$ ). These populated points  $i[\mathbf{m}]$  in  $(\hat{X}, \hat{P}, \hat{L})$  have the same state:

$$\hat{\mathbf{x}}_{i[\mathbf{m}]} = \mathbf{x}_i, \quad (\text{A27})$$

and periodically shifted positions:

$$\hat{\mathbf{p}}_{i[\mathbf{m}]} = \mathbf{p}_i + L(\mathbf{m} + \mathbf{c}_i) \quad \mathbf{p}, \quad (\text{A28})$$

where  $\mathbf{c}_i \in \mathbb{Z}^3$  is a constant chosen appropriately for each  $i$  so that  $\hat{\mathbf{p}}_{i[\mathbf{m}]}$  for all  $\mathbf{m} \in \mathbf{0} \leq \mathbf{m} \leq \mathbf{k} - \mathbf{1}$  reside within the unit cell region of  $\hat{L}$  (*i.e.*,  $\hat{L}^{-1}\hat{\mathbf{p}}_{i[\mathbf{m}]} \in [0, 1)^3$ ).

By simply applying Eq. (3) for  $(\hat{X}, \hat{P}, \hat{L})$ , we obtain

$$\hat{\mathbf{y}}_i = \frac{1}{Z_i} \sum_{j=1}^{k_1 k_2 k_3 N} \sum_{\mathbf{n}} \exp(\hat{\mathbf{q}}_i^T \hat{\mathbf{k}}_j / \sqrt{d_K} + \hat{\phi}_{ij(\mathbf{n})}) (\hat{\mathbf{v}}_j + \hat{\psi}_{ij(\mathbf{n})}), \quad (\text{A29})$$

where  $\hat{q}$ ,  $\hat{k}$ , and  $\hat{v}$  are key, query, and value of  $\hat{x}$  in  $\hat{X}$ . Scalar  $\hat{\phi}_{ij}(\mathbf{n})$  and vector  $\hat{\psi}_{ij}(\mathbf{n})$  encode the interatomic spatial relation as  $\phi(\hat{\mathbf{p}}_{j(\mathbf{n})} \quad \hat{\mathbf{p}}_i)$  and  $\psi(\hat{\mathbf{p}}_{j(\mathbf{n})} \quad \hat{\mathbf{p}}_i)$ , respectively, where  $\hat{\mathbf{p}}_{j(\mathbf{n})} = \hat{\mathbf{p}}_j + \hat{L}\mathbf{n}$ . We rewrite Eq. (A29) by rewriting populated point  $i$  as  $i[\mathbf{m}]$  and  $j$  as  $j[\mathbf{m}^\theta]$ , obtaining

$$\hat{\mathbf{y}}_{i[\mathbf{m}]} = \frac{1}{\hat{Z}_{i[\mathbf{m}]}} \sum_{j=1}^N \sum_{\mathbf{m}'} \sum_{\mathbf{n}} \exp(\hat{\mathbf{q}}_{i[\mathbf{m}]}^T \hat{\mathbf{k}}_{j[\mathbf{m}']} / \sqrt{d_K} + \hat{\phi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})) (\hat{\mathbf{v}}_{j[\mathbf{m}']} + \hat{\psi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})), \quad (\text{A30})$$

where  $\sum_{\mathbf{m}'}$  is short for  $\sum_{m'_1=0}^{k_1-1} \sum_{m'_2=0}^{k_2-1} \sum_{m'_3=0}^{k_3-1}$  with  $\mathbf{m}^\theta = (m_1^\theta, m_2^\theta, m_3^\theta)^T$ . Since  $\hat{x}_{i[\mathbf{m}]}$  in  $\hat{X}$  equals  $x_i$  in  $X$ , we substitute  $\hat{\mathbf{q}}_{i[\mathbf{m}]} = \mathbf{q}_i$ ,  $\hat{\mathbf{k}}_{j[\mathbf{m}']} = \mathbf{k}_j$ , and  $\hat{\mathbf{v}}_{j[\mathbf{m}']} = \mathbf{v}_j$  to obtain

$$\hat{\mathbf{y}}_{i[\mathbf{m}]} = \frac{1}{\hat{Z}_{i[\mathbf{m}]}} \sum_{j=1}^N \sum_{\mathbf{m}'} \sum_{\mathbf{n}} \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \hat{\phi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})) (\mathbf{v}_j + \hat{\psi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})). \quad (\text{A31})$$

Similarly to Eq. (4), this infinitely connected attention can be written in the following pseudo-finite periodic attention form:

$$\hat{\mathbf{y}}_{i[\mathbf{m}]} = \frac{1}{\hat{Z}_{i[\mathbf{m}]}} \sum_{j=1}^N \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \hat{\alpha}_{i[\mathbf{m}]j}) (\mathbf{v}_j + \hat{\beta}_{i[\mathbf{m}]j}), \quad (\text{A32})$$

where

$$\hat{\alpha}_{i[\mathbf{m}]j} = \log \sum_{\mathbf{n}} \sum_{\mathbf{m}'} \exp(\hat{\phi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})), \quad (\text{A33})$$

$$\hat{\beta}_{i[\mathbf{m}]j} = \frac{1}{\hat{Z}_{i[\mathbf{m}]j}} \sum_{\mathbf{n}} \sum_{\mathbf{m}'} \exp(\hat{\phi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})) \hat{\psi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n}), \quad (\text{A34})$$

and  $\hat{Z}_{i[\mathbf{m}]j} = \exp(\hat{\alpha}_{i[\mathbf{m}]j})$ . Here,  $\hat{\phi}_{i[\mathbf{m}]j[\mathbf{m}']}(\mathbf{n})$  encodes the interatomic spatial relation:

$$\hat{\mathbf{p}}_{j[\mathbf{m}']} \quad \hat{\mathbf{p}}_{i[\mathbf{m}]} = (\hat{\mathbf{p}}_{j[\mathbf{m}']} + \hat{L}\mathbf{n}) \quad \hat{\mathbf{p}}_{i[\mathbf{m}]} \quad (\text{A35})$$

$$= (\mathbf{p}_j + L\mathbf{m}^\theta + L\mathbf{c}_j \quad \mathbf{p} + \hat{L}\mathbf{n}) \quad (\mathbf{p}_i + L\mathbf{m} + L\mathbf{c}_i \quad \mathbf{p}) \quad (\text{A36})$$

$$= \mathbf{p}_j + (\hat{L}\mathbf{n} + L\mathbf{m}^\theta) + L(\mathbf{c}_j \quad \mathbf{m} \quad \mathbf{c}_i) \quad \mathbf{p}_i. \quad (\text{A37})$$

Notice here that

$$\hat{L}\mathbf{n} + L\mathbf{m}^\theta = (n_1 k_1 + m_1^\theta) \ell_1 + (n_2 k_2 + m_2^\theta) \ell_2 + (n_3 k_3 + m_3^\theta) \ell_3 \quad (\text{A38})$$

$$= L\mathbf{n}^\theta, \quad (\text{A39})$$

where  $\mathbf{n}^\theta = (n_1 k_1 + m_1^\theta, n_2 k_2 + m_2^\theta, n_3 k_3 + m_3^\theta)^T \geq \mathbb{Z}^3$ . Thus, we obtain

$$\hat{\mathbf{p}}_{j[\mathbf{m}']} \quad \hat{\mathbf{p}}_{i[\mathbf{m}]} = \mathbf{p}_j + L(\mathbf{n}^\theta + \mathbf{c}) \quad \mathbf{p}_i, \quad (\text{A40})$$

where  $\mathbf{c} = (\mathbf{c}_j \quad \mathbf{m} \quad \mathbf{c}_i) \geq \mathbb{Z}^3$ . Because  $\sum_{\mathbf{n}} \sum_{\mathbf{m}'}$  for terms of  $\mathbf{n}^\theta$  is equivalent to infinite series  $\sum_{\mathbf{n}'}$  over all  $\mathbf{n}^\theta \geq \mathbb{Z}^3$ , we obtain the following equivalence:

$$\hat{\alpha}_{i[\mathbf{m}]j} = \log \sum_{\mathbf{m}'} \sum_{\mathbf{n}} \exp(\phi(\hat{\mathbf{p}}_{j[\mathbf{m}']}(\mathbf{n}) \quad \hat{\mathbf{p}}_{i[\mathbf{m}]})) \quad (\text{A41})$$

$$= \log \sum_{\mathbf{n}'} \exp(\phi(\mathbf{p}_j + L(\mathbf{n}^\theta + \mathbf{c}) \quad \mathbf{p}_i)) \quad (\text{A42})$$

$$= \log \sum_{\mathbf{n}'} \exp(\phi_{ij}(\mathbf{n}' + \mathbf{c})). \quad (\text{A43})$$

Therefore,  $\hat{\alpha}_{i[\mathbf{m}]j}$  converges to the same value of  $\alpha_{ij}$  in Eq. (5), if they converge. Likewise,  $\hat{\beta}_{i[\mathbf{m}]j}$  converges to the same value of  $\beta_{ij}$  in Eq. (6). Provided that  $\hat{\alpha}_{i[\mathbf{m}]j} = \alpha_{ij}$  and  $\hat{\beta}_{i[\mathbf{m}]j} = \beta_{ij}$ , the right-hand side of Eq. (A32) equals the right-hand side of Eq. (4), thus proving  $\hat{\mathbf{y}}_{i[\mathbf{m}]} = \mathbf{y}_i$ .

At the overall architecture level, the above result indicates that applying the stacked self-attention blocks for an invariant representation, as  $\hat{X}^\theta = g(X, P, L, \mathbf{k}, \mathbf{p})$ , gives  $k_1 k_2 k_3$  duplicates of  $X^\theta = g(X, P, L)$ . For these  $\hat{X}^\theta$  and  $X^\theta$ , pooling operation  $h$  such as global mean or max pooling holds the invariance:  $h(\hat{X}^\theta) = h([X^\theta, \dots, X^\theta]) = h(X^\theta)$ . Thus, overall network transformation  $f$  is periodic invariant.

## G DIFFERENCE FROM POTNET

PotNet is a GNN-based crystal-structure encoder proposed by Lin et al. (2023). PotNet shares similarity with Crystalformer in that it uses infinite summations of interatomic potentials to model the periodicity of crystal structures. However, in addition to the obvious difference of being a GNN or a Transformer, *there is a more fundamental or conceptual difference in their representations of potential summations*. To see this, let us discuss how these methods perform state evolution.

**State evolution in PotNet.** Using our notation, the PotNet’s state evolution can be essentially expressed as

$$\mathbf{x}_i^l = g \left( \mathbf{x}_i, \sum_{j=1}^N f(\mathbf{x}_i, \mathbf{x}_j, \mathbf{e}_{ij}) \right), \quad (\text{A44})$$

where  $f$  and  $g$  are learnable GNN layers that perform message passing on all pairs of hidden state variables  $(\mathbf{x}_i, \mathbf{x}_j)$  and edge features  $\mathbf{e}_{ij}$  between them. Thus, PotNet presents itself in a standard GNN framework for fully connected graphs of nodes of unit-cell atoms. The most important part of PotNet is its edge feature  $\mathbf{e}_{ij}$ , which is an *embedding* of an infinite summation of scalar potentials. This summation is defined for each unit-cell atom pair  $(i, j)$  as

$$S_{ij} = \sum_{\mathbf{n}} (\kappa \mathbf{p}_{j(\mathbf{n})} \cdot \mathbf{p}^{\mathbf{n}}) \quad (\text{A45})$$

using a scalar potential function  $\kappa(r) : \mathbb{R}_+ \rightarrow \mathbb{R}$ . PotNet exploits three known forms of actual interatomic potentials, that is, Coulomb potential  $1/r$ , London dispersion potential  $1/r^6$ , and Pauli repulsion potential  $e^{-\alpha r}$ , via their summation:

$$\kappa(r) = \kappa_1/r + \kappa_2/r^6 + \kappa_3 e^{-\alpha r}, \quad (\text{A46})$$

where  $\kappa_1, \kappa_2, \kappa_3, \alpha$  are hyperparameters. Edge feature  $\mathbf{e}_{ij}$  is then provided as its embedding expanded and encoded via radial basis functions (RBF) and multi-layer perceptron (MLP) as

$$\mathbf{e}_{ij} = \text{MLP}(\text{RBF}(S_{ij})). \quad (\text{A47})$$

Here, we see that *PotNet exploits the infinite summations of known interatomic potentials to derive physics- and periodicity-informed edge features  $\mathbf{e}_{ij}$ , to be processed in a standard GNN framework*.

**State evolution in Crystalformer.** The core part of our state evolution is the attention operation provided in Eq. (3) and written again below:

$$\mathbf{y}_i = \frac{1}{Z_i} \sum_{j=1}^N \sum_{\mathbf{n}} \exp(\mathbf{q}_i^T \mathbf{k}_j / \sqrt{d_K} + \phi_{ij(\mathbf{n})}) (\mathbf{v}_j + \psi_{ij(\mathbf{n})}) \quad (\text{A48})$$

As described in Sec. 3.1, *this attention operation, as a whole, is interpreted as an infinite summation of interatomic potentials defined and performed in an abstract feature space*.

One may find some similarities of our  $\alpha_{ij}$  and  $\beta_{ij}$  with  $S_{ij}$  and  $\mathbf{e}_{ij}$  of PotNet, because they enable a standard fully-connected Transformer or a fully-connected GNN to incorporate the crystal periodicity. However, our  $\alpha_{ij}$  and  $\beta_{ij}$  are rather bi-products of the explicit potential summation in feature space performed as the infinitely connected attention, whereas  $S_{ij}$  is a potential summation calculated in potential space before the GNN message passing (typically as preprocessing) and  $\mathbf{e}_{ij}$  is an abstract embedding of  $S_{ij}$ .

### Summary.

- PotNet proposes a new type of physics-informed edge feature by using the values of infinite summations of interatomic potentials. This edge feature implicitly embeds the information of crystal periodicity and can thus allow a standard fully-connected GNN to be informed of crystal periodicity.
- Crystalformer deeply fuses the calculations of infinite potential summations with neural networks, by performing the infinitely connected attention as an infinite potential summation expressed in abstract feature space. This attention operation can be performed in a standard fully-connected Transformer by using proposed position encoding  $\alpha_{ij}$  and  $\beta_{ij}$ .

## H HYPERPARAMETERS FOR THE JARVIS-DFT DATASET

Compared to the Materials Project dataset, the JARVIS-DFT dataset is relatively small in both its training set size and average system size (*i.e.*, average number of atoms in unit cells, which is approximately 10 atoms in the JARVIS-DFT while 30 in the Materials Project). To account for these differences, we increase the batch size and number of total epochs as shown in Table A2.

Table A2: **Hyperparameters for the JARVIS-DFT dataset.**

	Form. energy	Total energy	Bandgap (OPT)	Bandgap (MBJ)	E hull
Training set size	44578	44578	44578	14537	44296
Batch size	256	256	256	256	256
Total epochs	800	800	800	1600	800

## I DETAILED ABLATION STUDIES

### I.1 PHYSICAL INTERPRETATIONS OF RESULTS IN TABLE 4

Table 4 indicates the inclusion of the value position encoding yields greater improvements for the formation energy and total energy while relatively limited improvements for the bandgap and energy above hull (E hull). This result can be understood from the fact that *the formation energy and total energy are absolute energy metrics, whereas the bandgap and energy above hull are relative energy metrics that evaluate gaps between two energy levels, as  $E = E_1 - E_0$ .*

The accurate prediction of such energy gaps is generally difficult, because it needs accurate predictions of both  $E_1$  and  $E_0$ . Furthermore, even when an algorithmic improvement in the prediction method leads to improvements in absolute energy prediction (e.g., formation energy prediction), it does not necessarily lead to improvements in energy gap prediction. Specifically, assume that an improved algorithm predicts energies  $E_1^0 = E_1 + \epsilon_1$  and  $E_0^0 = E_0 + \epsilon_0$  as better predictions than  $E_1$  and  $E_0$ , respectively. These improvements can directly lead to lower prediction errors of absolute energy metrics. However, when improvements  $\epsilon_1$  and  $\epsilon_0$  are made by the same mechanism, it may be possible that  $\epsilon_1 = \epsilon_0$  so that they are canceled out in energy gaps  $E^0 = E_1^0 - E_0^0$ .

We conjecture that the limited improvements in the bandgap and energy above hull prediction are due to these difficulties of energy gap prediction.

### I.2 CHANGING THE NUMBER OF SELF-ATTENTION BLOCKS

We further study the effects of the number of self-attention blocks. We here utilize the simplified variant because of its relatively good performance and high efficiency (Table 3). The results in Table A3 suggest that the performance moderately mounts on a plateau with four or more blocks.

Table A3: **Ablation studies by changing the number of self-attention blocks.**

Settings	# Blocks	Form. E.	Total E.	Bandgap (OPT)	Bandgap (MBJ)	E hull
Simplified	1	0.0664	0.0669	0.171	0.368	0.1313
	2	0.0627	0.0609	0.153	0.329	0.0965
	3	0.0558	0.0569	0.145	0.320	0.0676
	4	0.0541	0.0546	0.140	0.308	0.0517
	5	0.0525	0.0543	0.143	0.309	0.0484
	6	0.0503	0.0533	0.140	0.323	0.0430
	7	<b>0.0498</b>	<b>0.0515</b>	<b>0.131</b>	<b>0.298</b>	<b>0.0407</b>

Given the good performance of the seven-block setting, we test a seven-block variant of the proposed full model on the Materials Project and JARVIS-DFT datasets. The results in Tables A4 and A5 show that the larger model achieves slightly better accuracies, especially when the training set is large enough. Investigating how the performance changes with larger-scale datasets (Kirklin et al., 2015; Draxl & Scheffler, 2019) is left as a future topic.

Table A4: Performance of the larger model on the Materials Project (MEGNet) dataset.

	Formation energy 60000/5000/4239	Bandgap 60000/5000/4239	Bulk modulus 4664/393/393	Shear modulus 4664/392/393
4 blocks (default)	0.0186	0.198	0.0377	<b>0.0689</b>
7 blocks	<b>0.0185</b>	<b>0.184</b>	<b>0.0372</b>	0.0717

Table A5: Performance of the larger model on the JARVIS-DFT 3D 2021 dataset.

	Form. energy 44578/5572/5572	Total energy 44578/5572/5572	Bandgap (OPT) 44578/5572/5572	Bandgap (MBJ) 14537/1817/1817	E hull 44296/5537/5537
4 blocks (default)	0.0306	0.0320	0.128	<b>0.274</b>	0.0463
7 blocks	<b>0.0297</b>	<b>0.0312</b>	<b>0.127</b>	0.275	<b>0.0461</b>

## J RECIPROCAL SPACE ATTENTION FOR LONG-RANGE INTERACTIONS

### J.1 BACKGROUND

Before presenting our attention formulation in reciprocal space, we briefly introduce its background from physical simulations. In physical simulations, the difficulty of computing long-range interactions in periodic systems is well known. Given potential function  $\phi(r)$  and physical quantities  $\bar{v}_1, \dots, v_N \mathcal{G}$  (e.g., electric charges) associated with unit-cell atoms, these simulations typically involve the summation of interatomic interactions:

$$V_i = \sum_{j(\mathbf{n}) \neq i} (\phi_j(\mathbf{n}) - \phi_i) v_j, \quad (\text{A49})$$

where  $\sum_{j(\mathbf{n}) \neq i}$  sums for all the atoms  $j(\mathbf{n})$  in the structure, as  $\sum_{j=1}^N \sum_{\mathbf{n}}$ , except for  $j(\mathbf{n}) = i$ . Its direct computation expanding  $\mathbf{n}$  causes inefficiency for long-range  $\phi(r)$  due to slow convergence. Thus, DFT calculations often employ an efficient calculation method called Ewald summation.

In Ewald summation, potential  $\phi(r)$  is represented as the sum of short-range and long-range terms as

$$\phi(r) = \phi_{\text{short}}(r) + \phi_{\text{long}}(r). \quad (\text{A50})$$

In this manner, the summation of interactions is divided into two parts as

$$V_i = \sum_{j(\mathbf{n}) \neq i} \phi_{\text{short}}(\phi_j(\mathbf{n}) - \phi_i) v_j + \sum_{j(\mathbf{n}) \neq i} \phi_{\text{long}}(\phi_j(\mathbf{n}) - \phi_i) v_j. \quad (\text{A51})$$

The first part is the sum of fast-decay potentials in distance, which can be efficiently computed directly in real space. On the other hand, long-range potentials have concentrated low-frequency components and thus have fast decay in frequency. Therefore, the second part can be efficiently computed in reciprocal space via its Fourier series expansion.

### J.2 RECIPROCAL SPACE ATTENTION

Kosmala et al. (2023) recently proposed Ewald message passing in the GNN framework, which performs graph convolutions in both real and reciprocal space. We also import the idea of Ewald summation into our Transformer framework.

To obtain reciprocal-space representations of  $\alpha_{ij}$  and  $\beta_{ij}$  in Eqs. (5) and (6), it is sufficient to derive for infinite series  $\sum_{\mathbf{n}} \exp(\phi_{ij}(\mathbf{n})) \psi_{ij}(\mathbf{n})$  in the right-hand side of Eq. (6). We rewrite it as  $f(\mathbf{p}_j - \mathbf{p}_i)$  using the following 3D spatial function:

$$f(\mathbf{r}) = \sum_{\mathbf{n}} \exp(\phi_i(\mathbf{r} + n_1 \mathbf{l}_1 + n_2 \mathbf{l}_2 + n_3 \mathbf{l}_3)) \psi_i(\mathbf{r} + n_1 \mathbf{l}_1 + n_2 \mathbf{l}_2 + n_3 \mathbf{l}_3). \quad (\text{A52})$$

With this spatial function,  $\alpha_{ij}$  and  $\beta_{ij}$  are expressed as

$$\alpha_{ij} = \log f(\mathbf{p}_j - \mathbf{p}_i) \psi_i(\mathbf{p}_i), \quad (\text{A53})$$

$$\beta_{ij} = \frac{1}{Z_{ij}} f(\mathbf{p}_j - \mathbf{p}_i), \quad (\text{A54})$$

where  $f(\psi_i)$  denotes substitution  $\psi_i$  into Eq. (A52).

Since  $f$  is a periodic function, it can be alternatively expressed via Fourier series expansion with  $f$ 's Fourier coefficient  $F(\omega_m)$  as

$$f(\mathbf{r}) = \sum_{\mathbf{m} \in \mathbb{Z}^3} F(\omega_m) \exp(i\omega_m \cdot \mathbf{r}), \quad (\text{A55})$$

where  $i$  is the imaginary unit and

$$\omega_m = m_1 \mathbf{l}_1 + m_2 \mathbf{l}_2 + m_3 \mathbf{l}_3 \quad (\text{A56})$$

is a 3D vector analogous to the angular frequency of the Fourier transform.  $\omega_m$  is defined with three integers  $\mathbf{m} = (m_1, m_2, m_3) \in \mathbb{Z}^3$  with reciprocal lattice vectors:

$$(\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3) = \left( \frac{2\pi}{V} \mathbf{l}_2 \times \mathbf{l}_3, \frac{2\pi}{V} \mathbf{l}_3 \times \mathbf{l}_1, \frac{2\pi}{V} \mathbf{l}_1 \times \mathbf{l}_2 \right) \quad (\text{A57})$$

where  $V = (\mathbf{l}_1 \times \mathbf{l}_2) \cdot \mathbf{l}_3$  is the cell volume. Fourier coefficient  $F(\omega_m)$  is obtained via 3D integration:

$$F(\omega_m) = \frac{1}{V} \iiint_{\mathbb{R}_{\text{cell}}^3} f(\mathbf{r}) \exp(-i\omega_m \cdot \mathbf{r}) dx dy dz \quad (\text{A58})$$

$$= \frac{1}{V} \iiint_{\mathbb{R}^3} \exp(\phi_i(\mathbf{r})) \psi_i(\mathbf{r}) \exp(-i\omega_m \cdot \mathbf{r}) dx dy dz, \quad (\text{A59})$$

where  $\mathbb{R}_{\text{cell}}^3 \subset \mathbb{R}^3$  is the 3D domain of the unit cell. Intuitively,  $F(\omega_m)$  with small or large  $\|\omega_m\|$  represents a low or high frequency component of  $f(\mathbf{r})$ .

Computing  $\alpha_{ij}$  and  $\beta_{ij}$  with Eqs. (A53) and (A54) in reciprocal space using Fourier series in Eq. (A55) is effective when potential function  $\exp(\phi_{ij}(\mathbf{r}))$  is long-tailed, because  $F(\omega_m)$  in such a case decays rapidly as  $\|\omega_m\|$  increases.

### J.3 GAUSSIAN DISTANCE DECAY ATTENTION IN RECIPROCAL SPACE.

The reciprocal-space representation of  $\alpha_{ij}$  in the case of the Gaussian distance decay function in Eq. (7) is expressed as follows.

$$\alpha_{ij} = \log \left[ \frac{(2\pi\sigma_i^2)^{3/2}}{V} \sum_{\mathbf{m}} \exp\left(-\frac{\sigma_i^2 \|\omega_m\|^2}{2}\right) \cos(\omega_m \cdot (\mathbf{p}_j - \mathbf{p}_i)) \right] \quad (\text{A60})$$

Eq. (A60) is essentially an infinite series of Gaussian functions of frequencies  $\|\omega_m\|$ , which rapidly decay as frequencies increase. On the other hand, we omit  $\beta_{ij}$  in the reciprocal-space attention by assuming  $\psi_{ij}(\mathbf{r}) = \beta_{ij} = 0$ , because its analytic solution does not exist for our current  $\psi_{ij}(\mathbf{r})$  provided in Eq. (9). Exploring an effective form of  $\psi_{ij}(\mathbf{r})$  for the reciprocal-space attention is left as future work.

When  $\sigma_i^2 = \sigma_j^2$ , the two expressions of  $\alpha_{ij}$  with Eqs. (5) and (A60) become theoretically equivalent. Similarly to Ewald summation, however, we want to use the real-space attention for short-range interactions and the reciprocal-space attention for long-range interactions. To this end, we treat  $\sigma_i$  and  $\sigma_j$  as independent variables by following a parameterization method in Appendix C. Specifically, we parameterize  $\sigma_i^2$  as

$$\sigma_i^2 = r_0^2 \rho((\mathbf{q}_i^T \mathbf{w} - m)/s; a, b). \quad (\text{A61})$$

In this way,  $\sigma_i$  is lower bounded as  $\sigma_i > \sigma_{\text{lb}}$  with  $\sigma_{\text{lb}} = r_0 \sqrt{b}$ . We use  $(r_0, a, b) = (2.2, 0.1, 0.5)$  for  $\sigma_i$ . Now  $\sigma_i$  and  $\sigma_j$  have an upper and lower bound, respectively, as  $\sigma_i < \sigma_{\text{ub}} = 1.98\text{\AA}$  and  $\sigma_i > \sigma_{\text{lb}} = 1.56\text{\AA}$ . Therefore, *the combined use of the real-space and reciprocal-space attention can cover the entire range of interactions in principle*. We compute Eq. (A60) with a fixed range of  $\|\omega_m\| \leq 2$ . Note that because of the symmetry, we can reduce the range of  $m_1$  as  $0 \leq m_1 \leq 2$  by doubling the summed values for  $m_1 > 0$ .

Table A6: Effects of attention in the Fourier space on the JARVIS-DFT dataset.

Method	Form. energy	Total energy	Bandgap (OPT)	Bandgap (MBJ)	E hull
Crystalformer (real space)	<b>0.0306</b>	<b>0.0320</b>	0.128	<b>0.274</b>	0.0463
Crystalformer (dual space)	0.0343	0.0353	<b>0.126</b>	0.283	<b>0.0284</b>

#### J.4 RESULTS USING THE DUAL SPACE ATTENTION

As discussed in Sec. 6, we evaluate a variant of our method by changing half the heads of each MHA to the reciprocal attention provided in Appendix J.3. Table A6 shows the results with this dual-space attention.

We observe that the dual-space variant yields a significant improvement in the energy above hull prediction. Although its exact mechanism is unknown, we provide hypothetical interpretations below, as we do for results in Table 4 in Appendix I. In short, we conjecture that it is related to the fact that *the energy above hull is the only metric in Table A6 that involves the energy evaluation of different crystal structures than the targeted one.*

A more in-depth discussion needs to know the definition of the energy above hull. Let’s again express it as  $E = E_1 - E_0$ , where  $E_1$  corresponds to the formation energy of the target structure and  $E_0$  is the so-called convex hull provided by stable phases of the target structure. As we know, substances around us change their phases between solid, liquid, and gas depending on the temperature and pressure. Likewise, a crystal structure has multiple stable phases with different structures within the solid state. For example, a structure with a chemical formula of  $AB$  has phases that have different structures with chemical formulas of  $A_xB_{1-x}$ ,  $x \in [0, 1]$  (i.e., structures with the same element composition but different ratios of elements). When we plot  $y = \text{FormationEnergy}(A_xB_{1-x})$  for all possible phase structures of  $AB$ , we can draw a convex hull of the plots, as in Fig. A1, which will have some corner points that represent stable phases of  $AB$  (i.e., S1 to S4 in Fig. A1). The energy above hull is then defined as the vertical deviation between the formation energy of the target structure and the line of the convex hull, essentially representing the instability of the target structure.

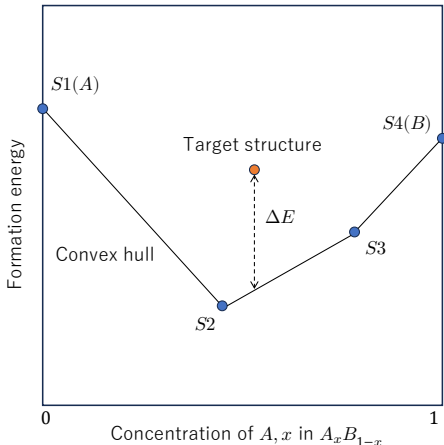


Figure A1: **Energy above hull.** Image courtesy of Ma, J., Hegde, V., Munira, K., Xie, Y., Keshavarz, S., Mildebrath, D., Wolverton, C., Ghosh, A., & Butler, W. (2017). *Computational investigation of half-Heusler compounds for spintronics applications.* *Phys. Rev. B*, 95, 024411.

When  $E_1$  and  $E_0$  involve the energy evaluation of different structures, it may be possible that the inclusion of long-range interactions has a positive impact on some of the stable phases of a given structure in  $E_0$  while not so for the given structure itself in  $E_1$ . This asymmetric relation between  $E_1$  and  $E_0$  may mitigate the aforementioned cancellation effect and may lead to improved prediction of  $E_0$  and  $E$ .