Examining graph neural networks for crystal structures: limitation on capturing periodicity

Sheng Gong\textsuperscript{1}, Tian Xie\textsuperscript{2,4}, Yang Shao-Horn\textsuperscript{1,3}, Rafael Gomez-Bombarelli\textsuperscript{1}, and Jeffrey C. Grossman\textsuperscript{1,*}

\textsuperscript{1}Department of Materials Science and Engineering, Massachusetts Institute of Technology, MA 02139, USA

\textsuperscript{2}Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology, MA 02139, USA

\textsuperscript{3}Department of Mechanical Engineering, Massachusetts Institute of Technology, MA 02139, USA

\textsuperscript{4}Microsoft Research, Cambridge, United Kingdom CB1 2FB
Abstract

Historically, materials informatics has relied on human-designed descriptors of materials structures. In recent years, graph neural networks (GNNs) have been proposed to learn the representations of crystal structures from the data end-to-end producing vectorial embeddings that are optimized for downstream prediction tasks. However, a systematic scheme is lacking to analyze and understand the limits of GNNs on capturing crystal structures. In this work, we propose to use human-designed descriptors as a bank of human knowledge to test whether black-box GNNs can capture knowledge of crystal structures behind descriptors from structures. We find that CGCNN and ALIGNN cannot capture periodicity of crystal structures well, and we thoroughly analyze the limitations of the GNN models that result in the failure from three aspects: local expressive power, long-range information, and readout function. We propose an initial solution, hybridizing descriptors with GNNs, to improve the prediction of GNNs for materials properties, especially phonon internal energy and heat capacity with 90% lower errors, and we analyze the mechanisms for the improved prediction. All the analysis can be easily extended to other deep representation learning models, human-designed descriptors, and systems such as molecules and amorphous materials.
Introduction

Recently, machine learning (ML) has been widely employed to high-throughputly predict properties of materials\textsuperscript{1-33}. Conversion of crystal structures into machine-readable numerical representations is one of the most critical steps for applications of ML in materials science\textsuperscript{34}. In general, there are two approaches to convert crystal structures into numbers: human-designed description and deep representation learning.

Human-designed descriptors are based on the collection of human understanding of compositions and structures of materials. Generally, people can easily understand the meaning of such descriptors. Mean electronegativity and difference of atomic radius of elements in materials are examples of compositional descriptors, and mean bond length and difference of coordination number of atoms in crystal structures are examples of structural descriptors of materials. Beyond simple descriptors, recently, researchers have proposed a series of descriptors for materials, such as Magpie\textsuperscript{26} compositional descriptors, classical force-field inspired descriptors (CFID)\textsuperscript{30}, Coulomb matrix\textsuperscript{35}, and fragment descriptors\textsuperscript{19}. Although ML models based on the human-designed descriptors have achieved some success in revealing the trend between human-understandable characteristics of materials and properties, these descriptors contain only information known to human-beings. Consequently, employing only descriptors to learn and predict materials properties might miss key structure-property relation unknown to human.

Deep representation learning, by definition, refers to the ML models that learn the numerical representation of materials automatically during the training of ML models. Although the learned representations are generally less understandable by human compared with human-designed descriptors, deep representation learning can uncover the pattern of structure-property relation unknown to human. Since materials can be intuitively represented as graphs, where
atoms forming the nodes and bonds the edges, graph neural networks (GNNs) have become the state-of-the-art deep representation learning method for materials science. SchNet$^{14}$ and CGCNN$^{29}$ are two classic GNN architectures designed for materials. They update the representations of each atom by the types of neighboring atoms and the bond lengths between atoms, and pool all the updated atom representations into an overall representation of each material. In later variants of GNN for materials such as iCGCNN$^{33}$, MEGNet$^{36}$ and GATGNN$^{37}$, bond representations are also updated during the convolution. Through multiple layers of graph convolutions, these models can implicitly encode many-body interactions. To explicitly encode many-body interactions, Gasteiger et al. proposed DimeNet$^{38}$ and GemNet$^{39}$ for molecules, and Choudhary et al. proposed ALIGNN for periodic crystal structures$^{27}$, where atom representations (one-body), bond representations (two-body) and bond angle representations (three-body) are all updated during the convolution via the construction of line graph (the nodes of the line graph are edges in the original graph, and edges of the line graph are angles between edges in the original graph). Together with other studies using higher-order information to improve expressiveness of GNN$^{40,41}$, ALIGNN-$d^{42}$, a recent variant of ALIGNN, updates the dihedral angle representation (four-body) by constructing line graph of line graph. Very recently, Batatia et al.$^{43}$ proposed a general formalism to encode local atomic environments by GNN with arbitrary body-order. Other efforts have also been made to improve GNN for crystal structures, such as inclusion of state attributes in MEGNet$^{36}$, attention mechanism in GATGNN$^{37}$, representations equivariant to rotations and inversion in E3NN$^{2,44}$, use of structure motifs in AMDNet$^{20}$, prediction of tensorial properties in ETGNN$^{45}$, and exploitation of correlations in spectral properties in Mat2Spec$^{1}$.

Although these variants of GNNs have achieved some success in learning materials properties, for capturing crystal structures, the improvements are mainly based on human intuition of local bonding environment, such as explicitly encoding bond angle (three-body
interaction) and dihedral angle (four-body) information, representations equivariant to rotations (orientations of bond vectors), and structure motifs. In general, prediction of materials properties is still challenging, and there is still no systematic approach and quantitative metric to analyze and understand the limitations of GNNs for crystal structures, especially for global information of crystal structures beyond local atomic environments.

In this work, we propose a systematic approach to analyze and quantify the limitations of GNNs for crystal structures, and propose a way to improve the GNNs models for predicting materials properties. As illustrated in Figure 1, we use the human-designed descriptors as a bank of human knowledge to test whether the current GNN models can capture certain knowledge of crystal structures. We test the GNNs by employing them to learn and predict the human-designed descriptors, and use the prediction accuracy as a quantitative metric for evaluation. The underlying assumption is that, if the model can accurately predict the descriptor, then the model can capture the knowledge behind the descriptor, otherwise the model cannot capture certain knowledge of crystal structures. We find that the GNNs are hard to capture periodicity of crystal structures, and we thoroughly analyze the limitation. We further hybridize the deep learning models with the human-designed descriptors, and test the descriptors-hybridized models on 13 important materials properties. We find that hybridization of GNNs and descriptors can result in up to 90% decrease of errors for predictions of phonon-related properties compared with original GNNs.
Figure 1. Schematic of analyzing whether a GNN can capture knowledge of crystal structures behind human-designed descriptors, and whether hybridization of GNN and human-designed descriptors can improve prediction performance for materials properties.

Results

Brief review of CGCNN and ALIGNN. In this work, we choose CGCNN and ALIGNN as two examples of GNNs to investigate their ability to capture human-designed descriptors, and as examples for improving prediction ability by hybridization with descriptors. CGCNN is one of the classic and most frequently used GNNs for materials, while ALIGNN is one of the state-of-the-art models for prediction of materials properties with the best performance on its in-house test set\textsuperscript{27} and the open Matbench test set\textsuperscript{46}. Both CGCNN and ALIGNN are specifically designed for predicting properties of periodic materials and have well-documented open-source codes to use and adapt. CGCNN explicitly encodes two-body interactions, and ALIGNN explicitly encodes three-body interactions. Although there are already GNNs that explicitly encode $n$-body interactions ($n \geq 4$)\textsuperscript{40-43}, they are not specifically designed for prediction of
properties of periodic crystal structures without comprehensive benchmark yet, therefore they are not examined in this work.

The architecture of CGCNN (https://github.com/txie-93/cgcnn) is summarized in equations (1) to (3):

\[
a_i^{(n+1)} = a_i^{(n)} + \sum_{j,k} \sigma(m_{(i,j)k}^{(n)} W_{\text{gate}}^{(n)}) \odot g(m_{(i,j)k}^{(n)} W_{\text{message}}^{(n)}) \quad \ldots \quad (1),
\]

\[
m_{(i,j)k}^{(n)} = a_i^{(n)} \oplus a_j^{(n)} \oplus b_{(i,j)k} \quad \ldots \quad (2),
\]

\[
\text{Output} = \text{AGG}(a_1^{(n^*)}, a_2^{(n^*)}, \ldots, a_N^{(n^*)}) \quad \ldots \quad (3).
\]

Here, \(a_i^{(n)}\) denotes the representation of atom \(i\) at layer \(n\), \(b_{(i,j)k}^{(n)}\) representation of the \(k^{th}\) bond between atom \(i\) and \(j\) at layer \(n\), \(n^*\) the final convolution layer, \(W_{\text{gate}}^{(n)}\) the gate matrix at layer \(n\), \(W_{\text{message}}^{(n)}\) the message matrix at layer \(n\), \(m_{(i,j)k}^{(n)}\) the message from atom \(j\) to atom \(i\) via the \(k^{th}\) bond, \(\odot\) element-wise multiplication, \(\oplus\) concatenation, \(\sigma\) and \(g\) non-linear activation functions, \(\text{AGG}\) the aggregation (readout) function. In CGCNN, the implemented aggregation function can be written as:

\[
\text{Output} = \text{FCN}(\frac{1}{N_a} \sum_{i=1}^{N_a} a_i^{(n^*)}) \quad \ldots \quad (4),
\]

where the output is calculated by first taking the average of all atom representations, then feeding the averaged representation to a fully connected network. The reason for using average pooling (equation (4)) is that intensive materials properties, such as band gap and refractive index, are invariant to the (supercell) size of crystal structures. In summary, in each convolution layer, CGCNN uses neighboring atoms and bond lengths as messages to each atom, and updates each atom representation by feeding the messages into a gate layer and a message processing layer. After convolutions, CGCNN pools all atom representations by taking the average and input the pooled material representation into a fully connected network to compute
The architecture of ALIGNN (https://github.com/usnistgov/alignn) is summarized in equations (5) to (10), with equations (5) to (7) describing the atomistic graph, and equations (8) to (10) the line graph:

\[
a_i^{(n+1)} = a_i^{(n)} + g(a_i^{(n)}W_{self}^{(n)} + \sum_{j,k} g'(b_{(i,j)k}^{(n)}W_{message}^{(n)}a_j^{(n)}) \]  

\[
b_{(i,j)k}^{(n+1)} = b_{(i,j)k}^{(n)} + g(m_{(i,j)k}^{(n)}W_{gate}^{(n)}) \]  

\[
m_{(i,j)k}^{(n)} = a_i^{(n)} \oplus a_j^{(n)} \oplus b_{(i,j)k}^{(n)} \]  

\[
b_i^{(n+1)} = b_i^{(n)} + g(b_i^{(n)}W_{self}^{(n)} + \sum_{j} g'(t_{(i,j)k}^{(n)}W_{message}^{(n)}b_j^{(n)}) \]  

\[
t_{(i,j)k}^{(n+1)} = t_{(i,j)k}^{(n)} + g(m_{(i,j)k}^{(n)}W_{gate}^{(n)}) \]  

\[
m'_{(i,j)k}^{(n)} = b_i^{(n)} \oplus b_j^{(n)} \oplus t_{(i,j)k}^{(n)} \]  

Here, \(t\) denotes the representation of bond angle, and other symbols share similar meaning to that of CGCNN. In summary, in each convolution layer, ALIGNN updates atom representations by neighboring atoms and bonds, updates bond representations twice: by connected atoms, and by neighboring bonds and bond angles, and update bond angle representations by connected bonds. After convolutions, ALIGNN uses average pooling in equation (4) to collect atom representations as the material representation, and calculate the property by a feed-forward network.

For building periodic crystal graphs, in their default settings, both CGCNN and ALIGNN use a cut-off radius of 8 Å for 12 nearest neighbors, and both of them use radial basis functions to expand the interatomic distances for initialization of bond representations. ALIGNN also uses radial basis functions to expand cosines of bond angles for initialization of bond angle
representations. CGCNN updates atom features by 3 graph convolution layers, and ALIGNN updates atom features by 4 line graph convolution layers (equations (5) to (10)) and 4 normal graph convolution layers (equations (5) to (7)). In the following, we use the CGCNN and ALIGNN with the default setting unless specifically mentioned.

Figure 2. Learning and predicting human-designed descriptors to examine whether the GNNs can capture certain human knowledge. a and b $R^2$ scores of predictions of human-
designed structural descriptors from CGCNN, ALIGNN and ROOST for local and global structural descriptors, respectively. The full names of the descriptors are listed in Table 1.

**Learning and predicting human-designed descriptors.** In this section, we employ CGCNN and ALIGNN to learn and predict structural descriptors of a subset of crystal structures in the Materials Project database\(^4^7\) (“MP dataset” as below; details in the Methods section) to examine the ability of the GNNs to capture certain knowledge behind the descriptors. As a baseline, we also use ROOST\(^4^8\), one of the most powerful composition-only deep learning models, to learn and predict the structural descriptors.

In Figure 2a, we show the accuracies of predictions of some of the most basic local structural descriptors calculated by matminer\(^4^9\) from CGCNN,ALIGNN, and ROOST in terms of \(R^2\) scores (\(R^2 = 1 - \frac{\sum(y_i - y_i,\text{true})^2}{\sum(y_i,\text{true} - \bar{y})^2}\), \(y_i\) predicted value, \(y_i,\text{true}\) true value, \(\bar{y}\) mean of true values). We can see that, for most local structural descriptors, both CGCNN and ALIGNN can properly predict them with \(R^2\) scores close to or higher than 0.8, and both of the two structure-based models outperform the composition-only model (ROOST). Because local descriptors in this work are essentially statistics of local environments around each atom, the explicit encoding of bond angles (three-body interaction) in ALIGNN might explain why ALIGNN outperforms CGCNN for learning local structural descriptors as in Figure 2a. The cases with lower \(R^2\) scores in Figure 2a, such as max_rela_bond_len (maximum relative bond length) and std_avg_bond_ang (standard deviation of average bond angles), can be attributed to the fact that, average pooling (equation (4)) is used by both CGCNN and ALIGNN to obtain the mean statistics of atom representations, while the two descriptors here describe the maximum and standard deviation of a collection of atomic environments.
In addition to basic local descriptors, we also test the ability of CGCNN and ALIGNN to capture knowledge behind more global structural descriptors. In Figure 2b, we show the accuracies of predictions of some of the most basic global structural descriptors calculated by matminer\textsuperscript{49} and pymatgen\textsuperscript{50} from CGCNN, ALIGNN, and ROOST. Both CGCNN and ALIGNN can predict density, vpa (volume per atom), packing fraction, and natoms (number of atoms in the primitive cell; in this work, the “primitive cell” is defined as the Niggli reduced cell\textsuperscript{51, 52}) with $R^2$ scores close to or higher than 0.8. However, they cannot predict struct_comp_cell (structural complexity per cell\textsuperscript{53}) and lattice constants ($a$, $b$, $c$, $\alpha$, $\beta$, $\gamma$; in this work, $a$ denotes the length of the longest lattice vector, $c$ the shortest, and $\alpha$ denotes the largest lattice angle, $\gamma$ the smallest) well. Both structure-based models outperform the composition-only model, and ALIGNN outperforms CGCNN, except $\alpha$ and $\gamma$. 
**Figure 3. Limitation of GNNs on capturing periodicity.** a Illustration of the receptive field of an atom in a GNN and periodicity of a 1-dimensional (1D) structure. Here, atom $i$ receives information from atoms 1 to $N$, and two cases of periodicity are plotted: the short periodicity from atom 1 to 3 and the long periodicity from atom 1 to $N+1$. b Illustration of 1D single carbon chains as toy structures. The chains are along the $x$ direction with periodicity, with random displacement of each atom in the $y$ and $z$ directions. c Illustration of 1D chains with zigzag and armchair configuration, respectively. d Illustration of 1D double chain. e and f $a_{\text{true}}$ versus $a_{\text{pred}}$ of the datasets of 1D short chains and 1D long chains from default CGCNN, respectively. g $R^2$ scores of predictions of $a$ of 1D short chains and 1D long chains, and $a, b, c$ of the MP dataset, from default CGCNN, CGCNN with 8 convolution layers, and CGCNN connecting 18 nearest neighbors within 12 Å, respectively. h $R^2$ scores of prediction of $a$ of 1D short chains and 1D long chains, and $a, b, c$ of the MP dataset from CGCNN with average pooling and CGCNN with sum pooling, respectively.

**Limitations of GNNs on capturing periodicity.** Although previous works have suggested that lattice constants of crystal structures are learnable based on only compositions\textsuperscript{54, 55}, the results in this work show that even with structures as input, CGCNN and ALIGNNN cannot capture lattice constants well. In this section, we thoroughly analyze the possible reasons for such failure and obtain insights for improving GNNs for crystal structures.

Lattice constants describe the periodicity of atomic structures. If $A(r)$ describes the type of atom at position $r$ (“none” if there is no atom at that position), and if $R$ is a linear combination of lattice vectors, then periodicity requires that:

$$A(r) = A(r + R)......(11).$$

In the 3-dimensional (3D) space, we need 3 linearly independent lattice vectors to describe the periodicity of atomic structures. Lattice constants describe the periodicity by the lengths of lattice vectors ($a, b, c$) and angles between lattice vectors ($\alpha, \beta, \gamma$). To simplify the analysis, in addition to 3D crystal structures in the MP dataset, we also consider the toy cases of quasi-
1D atomic chains as in Figure 3a, where periodicity is imposed only along the $x$ direction and no constraint is imposed along the other two directions. In this quasi-1D space, we only need the length of the lattice vector ($a$) to describe the periodicity: $A(r) = A(r + a)$.

For GNNs with average pooling in equation (4), since they use the average local atomic environments to represent the atomic structures, they capture periodicity by learning how equation (11) affects the local atomic environments within the receptive fields of atoms in the GNNs. Receptive field of each atom describes the range of the space where information can be propagated to the atom through the GNNs, and it depends on the number of neighbors each atom can connect to and the number of convolution layers in the GNNs:

$$\text{range of receptive field} \propto \text{number of neighbors} \times \text{number of convolutions} \ldots \ldots \text{(12)}.$$  

If the length of the periodicity (length of lattice vector) is smaller than the lengths of the receptive fields of atoms in the GNNs, then the GNNs might be able to capture the short periodicity; however, if the length of the periodicity is larger than the lengths of the receptive fields of atoms, then in principle the GNNs cannot capture the long periodicity. For example, as in Figure 3a, if the periodicity is short, such as the top red arrow which requires that atom 1 and atom 3 (atom $n$ and atom $n+2$) have the same type and coordinates in the $y$ and $z$ directions, then the local atomic environment input to atom $i$ is constrained by such periodicity, and the GNNs might be able to capture the constraint and periodicity. However, if the periodicity is long, such as the bottom red arrow describing that the periodicity is imposed between atom 1 and atom $N+1$ (one atom beyond the receptive field), then there is no constraint inside the receptive field of atom $i$, and the GNNs cannot capture the long constraint and periodicity.

To analyze the behaviors of GNNs on capturing periodicity, in this section, we introduce toy datasets of quasi-1D carbon chains as illustrated in Figure 3b (“1D dataset” as below; details in the Methods section), and we create two versions of the 1D datasets: a short dataset where
the periodicity of each chain is shorter than the receptive fields of atoms (1D, short), and a long dataset where the periodicity is longer than the receptive fields (1D, long). We use the default CGCNN to learn and predict the length of lattice vector ($a$) of the two datasets, and in Figure 3e and 3f, we show the predicted $a$ versus true $a$ of the two datasets. We can see that, for the short chains, CGCNN can predict $a$ well with the $R^2$ score larger than 0.8, while for the long chains, CGCNN cannot predict $a$ well. The prediction results of $a$ of the quasi-1D carbon chains support our analysis above that GNNs might be able to capture short periodicity while hard to capture long periodicity.

Although the periodicity of most short chains in this work can be properly learned as in Figure 3e, theoretically, GNNs with limited local expressive power is not able to fully determine the periodicity. Since Chen et al.\textsuperscript{56} have proved the equivalence between the ability of GNNs to distinguish graphs and approximate graph functions, if a GNN cannot distinguish two atomic graphs with different periodicity, then the GNN cannot fully determine the graph function describing the periodicity. In Figure 3c, we show two cases of 1D chains: 1D zigzag chain and 1D armchair chain, which represent structure prototypes of some real crystal structures such as organic crystals\textsuperscript{57} and metal chalcogenides\textsuperscript{58}. If a GNN uses only diatomic distances to encode local atomic environments (such as CGCNN), and if the GNN only connect to the nearest neighbors (1 to 2), then the GNN cannot distinguish different zigzag and armchair 1D chains with the same bond length but different bond angles and cannot capture the angle dependence of $a$. If the GNN can connect to the second nearest neighbors (1 to 3), then the GNN is able to distinguish zigzag and armchair 1D chains with different bond angles; however, it is still not able to distinguish between zigzag and armchair chains with the same bond length and bond angle. The analysis suggests that, to improve the ability of GNNs to capture short periodicity, it might be helpful to increase the local expressive power of GNNs to distinguish structures with different periodicity.
In Figure 3g, we show the effects of number of convolution layers and number of neighbors of CGCNN on capturing periodicity of 1D chains and 3D crystal structures. As in equation (12), both increasing number of convolution layers and increasing number of neighbors extend receptive fields of atoms in CGCNN, and as the discussion of zigzag and armchair chains above, increasing number of neighbors can lead to higher local expressive power to distinguish graphs. From Figure 3g, we can see that for short chains, increasing number of neighbors leads to better prediction of $a$, which supports our suggestion above that improving the local expressive power can help to capture short periodicity, while increasing number of convolution layers results in worse prediction of $a$, which might be because deeper GNNs are harder to train\textsuperscript{59, 60}. For long chains, both increasing number of neighbors and number of convolution layers result in better prediction of $a$, indicating that extending the receptive fields of atoms in CGCNN can help to capture long periodicity. As for lengths of lattice vectors of real 3D structures in the MP dataset (mixed with short and long structures), we can see that both increasing number of neighbors and number of convolution layers lead to better prediction of $a, b, c$. However, we find that increasing number of convolution layers by 133% and number of neighbors by 50% just lead to moderate improvement of prediction of $a, b, c$. Since the cost of graph convolution operations is proportional to number of convolution layers and neighbors, we suggest that simply increasing number of convolution layers and neighbors might not be an ideal way to improve the ability of GNNs to capture periodicity.

The analysis above is based on average pooling in equation (4). If we use sum pooling in equation (13) with size extensibility:

$$\text{Output} = \sum_{i=1}^{N_a} \text{FCN}(a_i^n) \quad \ldots \quad (13),$$

then the GNNs capture periodicity by summing contributions of each atom to the lattice vectors. In Figure 3h, we show the $R^2$ scores of predictions of $a$ of the 1D chains and natoms, $a, b, c$
of the MP dataset from CGCNN with average pooling and sum pooling, respectively. For 1D short chains, sum pooling can lead to better prediction of \( a \) than average pooling, which might be explained by the fact that sum pooling is more expressive than average pooling\(^{61}\). For 1D long chains, sum pooling can result in significantly better prediction of \( a \) than average pooling, because average pooling requires that each atom encodes information from one end of the long primitive cell to the other end to capture the structural constraint imposed by the periodicity, while sum pooling needs only local contributions of each atom to the lattice vectors. Consistent with the results of \( a \) of the 1D chains, for \( n \) atoms, \( a, b, c \) of the MP dataset, sum pooling can also result in better prediction than average pooling. The stronger ability of sum pooling to capture periodicity might lead to better prediction of extensive materials properties, and in Figure S1 in the Supplementary Information, we show that sum pooling can provide better prediction than average pooling for phonon internal energy \((U)\), phonon heat capacity \((C_v)\) and magnetization \((M)\).

Despite the improvement, we suggest that sum pooling is not an ideal solution to the problem of hard to capture periodicity. Periodicity and lattice constants of the primitive cells do not scale with supercell size and are intensive characteristics of crystal structures. In principle, sum pooling cannot be employed in machine learning of materials’ intensive properties due to the requirement of (supercell) size invariance\(^{29}\). The improvement of sum pooling over average pooling in Figure 3h and S1 is based on the fact that primitive cells of crystals are used as input to the GNNs in this work. Even if only primitive cells are input to the GNNs, sum pooling might also fail to capture periodicity in some cases, as periodicity does not always scale with the number of atoms in the primitive cells. For example, in Figure 3d we show the case of 1D double chains. Compared with 1D single chains in Figure 3b and 3c, 1D double chains can have similar periodicity but twice number of atoms. In Figure S2, we show that, compared with the datasets with only 1D single chains, sum pooling is less powerful to capture the periodicity
of the datasets mixed with 1D single and double chains.

From Figure 2b, we can see that ALIGNN can have better prediction of natoms, $a$, $b$, $c$ of the MP dataset. The better predictions might result from two aspects: on the one hand, ALIGNN has stronger local expressive power than CGCNN as it explicitly encodes bond angles, and on the other hand, ALIGNN has larger receptive field than CGCNN, as in each convolution layer in CGCNN, a node receives messages only from the first shell of bonds and neighbors in equation (2), while in each convolution layer in ALIGNN, a node also receives messages from the second shell of bonds in equation (10). Although default ALIGNN has 8 convolution layers while default CGCNN has only 3 convolution layers, from Figure 3g we can see that increasing the number of convolution layers of CGCNN to 8 leads to only moderate improvement and cannot make the predictions of $a$, $b$, $c$ from CGCNN as accurate as that of ALIGNN, which shows that different number of convolution layers in default CGCNN and ALIGNN is not a critical factor on their relative ability to capture periodicity of the MP dataset.

In Figure 2b and S3, we show that both CGCNN and ALIGNN cannot learn the lattice angles of the primitive cell well, and sum pooling, more convolutions, and more neighbors cannot improve the prediction. Here we partially attribute the phenomenon to the artificial choice of lattice angles. More discussions about determination of primitive cell are provided in the Supplementary Information. In this work, we choose the set of six parameters $(a, b, c, \alpha, \beta, \gamma)$ as a widely used rotationally invariant representation of lattice vectors, which might add artificial difficulty to the learning of periodicity. For example, in addition to the problems associated with learning and prediction of $a$ of 1D cases as above, for learning and prediction of the length of the longest lattice vector of 3D structures, the GNNs need to first identify which dimension is associated with the largest length, then determine the largest lattice length. For fairer evaluation, it is necessary to develop representations of periodicity that are equivariant to rotations to avoid the mentioned additional difficulty.
According to MLatticeABC\textsuperscript{54} and CRYSPNet\textsuperscript{55}, lattice constants of high-symmetry materials are reported to be learnable based on only compositions of materials, while here we show that lattice constants are not learnable by the GNNs even with structures as input. The difference between this work and the two previous works is that, in the previous works, materials with different symmetry are learned separately, and lattice constants of high-symmetry materials are reported to be more learnable than that of low-symmetry materials, while in this work the MP dataset is mixed with different symmetries and is biased to materials with low symmetry. More details about the MP dataset are provided in the Methods section.

In this section, we discuss the limitations of the GNNs on capturing periodicity mainly in three aspects: limited local expressive power, difficulty of capturing long-range information beyond receptive fields of atoms, and average pooling as the readout function. For local expressive power, advancements of GNNs to capture more structural characteristics, such as ALIGNN-$d$ for dihedral angles\textsuperscript{42} and equivariant representations for orientation of bond vectors\textsuperscript{2, 44}, might be helpful to better capture periodicity of structures with lattice vectors shorter than the receptive fields. For long-range information, on the one hand efforts to train very deep GNNs effectively and efficiently, such as DeeperGATGNN\textsuperscript{62}, are helpful to extend the receptive fields of atoms, on the other hand, the idea of topological message passing\textsuperscript{63, 64} might be useful to capture long-range information by connecting nodes in the same cell complex that are far from each other, and the idea of Implicit Graph Neural Networks (IGNN)\textsuperscript{65} might also be useful to bypass the problems associated with training very deep graph neural networks by obtaining implicitly defined state vectors from a fixed-point equilibrium equation. It is also necessary to further develop readout function to collect the long-range information with size invariance, and the whole-graph self-attention based readout function used in GraphTrans\textsuperscript{66} might be a good starting point to collect global information of crystals.
**Initial solution: descriptors-hybridized deep representation learning.** From the results of learning human-designed descriptors, we know that GNNs might not capture all knowledge behind human-designed descriptors. One way to overcome the issue is to design better GNN architectures for specific information, such as the long-range information. Another way to overcome the issue is to input the missing knowledge into the deep representation learning models. Although this idea is straightforward and used in previous works\(^3,67\), such as the incorporation of lattice vectors in GeoCGNN\(^3\), the previous works did not explain the role of the additional information with quantitative evidence. In this section we show the mechanisms of how inputting certain knowledge to GNNs improves prediction of materials properties, and we find that the hybridization with descriptors can lead to huge improvement for prediction of some materials properties, especially vibrational properties that largely depend on periodicity.

We construct the descriptors-hybridized graph neural networks as below:

\[
\text{Output} = \text{FCN}(\frac{1}{N_a} \sum_{i=1}^{N_a} \alpha_i^n \oplus \text{descriptors}) \quad \ldots \quad (14). 
\]

In other words, we concatenate the vector of descriptors to the vector of learned representation, and input the hybridized representation vector to the fully-connected network.

In Figure 4a, we show the prediction results of descriptors-hybridized CGCNN and ALIGNN (de-CGCNN and de-ALIGNN) on 13 materials properties, with the full names of the abbreviations of properties in Table 1, and detailed errors in the Table S1. The set of properties includes final energy \((E_{\text{fin.}})\), band gap \((E_g)\), bulk and shear modulus \((K \text{ and } G)\), lattice thermal conductivity \((\kappa)\), phonon internal energy and heat capacity at 300K \((U \text{ and } C_v)\), Poisson ratio \((\nu)\), modulus of the piezoelectric tensor \((||e||_\infty)\), electronic and total dielectric constant \((\varepsilon_e \text{ and } \varepsilon_t)\), refractive index \((n)\) and total magnetization \((M)\). The errors of the machine learning models are presented using the metric 

\[
\frac{\Sigma |y_i - y_{i,\text{true}}|}{\Sigma |y_{i,\text{true}} - \bar{y}|},
\]

which is invariant to scaling and used
in the ALIGNN paper. Typically, a model with the MAE/MAD smaller than 0.2 is thought to be a good predictive model. We can see that de-CGCNN has improved prediction performance for most properties compared with the original CGCNN, and de-ALIGNN has close-to or larger than 10% improvement for four properties ($\kappa$, $U$, $C_v$, and $M$) and similar performance for other properties compared with the original ALIGNN. Both de-CGCNN and de-ALIGNN outperform the descriptors-only model for all properties, regardless of whether CGCNN and ALIGNN outperform the descriptors-only model.

Figure 4. Prediction performance of descriptors-hybridized GNNs. a MAE/MAD ratio of predictions of 13 materials properties from machine learning models based on only descriptors, CGCNN, ALIGNN, and their descriptors-hybridized version (de-CGCNN and de-ALIGNN).
b, c and d Relative feature importance of representations from de-CGCNN for $C_v$, $\kappa$, and $M$, respectively. e Ratio of feature importance of input human-designed descriptors to the total feature importance from de-CGCNN for the 13 materials properties.

Providing missing knowledge to the GNNs might lead to huge improvement. In Figure 4a, we observe that both de-CGCNN and de-ALIGNN have huge improvement for the prediction of $U$ and $C_v$, with around 90% lower errors compared with CGCNN and ALIGNN, respectively. To understand the improvement, we show the feature importance spectrum of de-CGCNN for prediction of $C_v$ in Figure 4b. We can see that, the human-designed descriptors play important roles in learning $C_v$, with $a$ being the most important feature, while the learned features are much less important. Therefore, the poor prediction ability of CGCNN and ALIGNN for $U$ and $C_v$ can be explained by the fact that $a$ is important to the two properties but CGCNN and ALIGNN cannot learn $a$ well as above. The distribution of feature importance agrees well with the phenomenon in Figure 4a that, using the machine learning model based on only human-designed descriptors can have lower errors for prediction of $U$ and $C_v$ compared with the GNNs.

The importance of the input human-designed descriptors to $U$ and $C_v$ can be justified physically as below. Approximately, if we only consider the acoustic phonons (collective vibrations for all atoms in the primitive cell), according to the Debye model of density of states, the phonon internal energy ($U$) and heat capacity ($C_v$) per primitive cell can be written as:

\[
U \approx 9k_B T \left( \frac{T}{\Theta} \right)^3 \int_0^{x_D} dx \frac{x^3}{e^{x} - 1} \ldots (15),
\]

\[
C_v = \left( \frac{\partial U}{\partial T} \right)_v \approx 9k_B \left( \frac{T}{\Theta} \right)^3 \int_0^{x_D} dx \frac{x^4 e^x}{(e^x - 1)^2} \ldots (16),
\]

\[
x_D \equiv \frac{\Theta}{T} \ldots (17),
\]
\[ \theta = \frac{\hbar}{k_B} \left( \frac{6\pi^2}{V} \right)^{\frac{1}{3}} \] ......(18),

where \( \theta \) is the debye temperature, \( V \) is the volume of the primitive, and \( v \) is the velocity of sound, which can be approximated by the first-order Hooke’s law:

\[ v \approx \sqrt{\frac{C}{m}d} ......(19), \]

where \( C \) is the effective spring constant, \( m \) is the mass of atoms in the primitive cell, and \( d \) is the effective distance between atomic planes along the direction of vibration. Therefore, with the information of \( V \), \( C \), \( m \), and \( d \), we can estimate acoustic \( U \) and \( C_v \) per primitive cell at given \( T \) within the Debye model. Since the set of descriptors in this work includes density and lattice constants, the information of \( V \), \( m \), and \( d \) can be directly obtained by machine learning models from the input descriptors. For \( C \), because it is related to the bonding strength, it can be estimated by the bond length-related descriptors. Consequently, machine learning models based on the set of descriptors in this work can approximate \( U \) and \( C_v \) well within the Debye model, which explains why machine learning model based on only descriptors outperforms CGCNN and ALIGNN in Figure 4a, as CGCNN and ALIGNN cannot estimate lattice constants well as in Figure 2b. More discussions about the relation between this work and the prediction of \( U \) in Legrain et al.\textsuperscript{69} and the prediction of phonon density of states by E3NN\textsuperscript{2} and Mat2Spec\textsuperscript{1} are provided in the Supplementary Information.

\( \kappa \) and \( M \) are another two properties with around 10% improvement from both de-CGCNN and de-ALIGNN. It is known that \( \kappa \) depends significantly on periodicity\textsuperscript{70}, and as shown in Figure 4c, some input descriptors, including \( b \), are important to the prediction of \( \kappa \). As for \( M \), as shown in Figure 4d, some descriptors like structural complexity and lattice constants contribute to the prediction of \( M \). In Figure 4e, we show the ratio of feature importance from the human-designed descriptors to the total feature importance from de-CGCNN. We can see
that most properties without significant improvement in Figure 4a have low contributions from
the input human-designed descriptors, with the exception of $v$ and $\|e\|_\infty$ where all the models
perform poorly. The phenomenon that hybridization with descriptors has larger improvement
for CGCNN than ALIGNN might be explained by the fact that, CGCNN captures these
descriptors worse than ALIGNN as in Figure 2, therefore hybridization with descriptors
provides more missing information to CGCNN than ALIGNN.

In addition to providing missing information, hybridization with descriptors might also have
other impacts on the GNNs. In the Supplementary Information, we show that hybridization of
descriptors can bias the learned representations less correlated with the input descriptors,
although how such bias affects prediction performance is not clear yet. Another two questions
worth being further studied are, how does the improvement scale with dataset size, and how to
choose the set of input descriptors for optimal performance. Are the two mentioned behaviors
(scaling and selection of descriptors) similar with or different from that of the descriptors-only
models and pure deep representation learning models?

**Discussions and Conclusions**

In summary, we propose a systematic approach to analyze the representation power of GNNs
on crystal structures. We use human-designed descriptors as a bank of human-knowledge to
test whether CGCNN and ALIGNN can capture knowledge of crystal structures behind
descriptors. We find that both GNNs can capture basic local structural descriptors well, but
cannot capture periodicity of crystal structures. We analyze the limitations of the GNNs on
capturing periodicity from three perspectives: local expressive power, long-range information
and pooling function. We also test the idea of hybridization with descriptors to improve the
performance of GNN, and show that descriptors-hybridized CGCNN and ALIGNN have better
prediction performance for some materials properties than the original CGCNN and ALIGNN, especially phonon internal energy and phone heat capacity with 90% lower errors.

As a starting point, the analysis performed in this work can be easily extended to other deep representation learning models, human-designed descriptors, and systems beyond crystals such as molecules and amorphous materials. This work shows that the fields of human-designed descriptors and deep representation learning can be developed synergically. For new deep representation learning models, their representation power on crystal structures can be tested by learning existing human-designed descriptors, and for new descriptors, they can be used to reveal how well the existing deep representation learning models capture the knowledge behind these descriptors, and they can be hybridized with deep representation learning models for improved prediction performance. We hope this work can inspire further development of deep representation learning, human-designed descriptors and hybridized machine learning models for crystal structures and materials science.

**Methods**

**Datasets.** In this work, we choose 25 (in Figure 2) human-designed descriptors to test their learnability to CGCNN and ALIGNN, and hybridize 29 descriptors (all descriptors in Table 1) with the two GNNs to test the prediction performance. The list of descriptors is provided in Table 1, with descriptors after $\gamma$ included in the second task but not in the first task. The criterion for choosing the 25 descriptors in the first task is that they are easy to understand and easy to obtain from crystal structures, and the reason for not testing coordination number (CN) in the first task is that we anticipate that CN can be learned well given the definition of GNN, and the reason for not testing symmetry in the first task is that we know symmetry cannot be learned well as lattice constants cannot be captured. Number of atoms and lattice constants of
the primitive cell are determined by the Niggli reduction implemented in the Structure class in pymatgen, and other descriptors are calculated by Matminer. For the descriptor “standard deviation average bond length” (and similar descriptors), the calculation procedure is first calculating average bond length for each atom, then calculating the standard deviation for the average bond length of all atoms.

In this work, most real 3D crystal structures (primitive cells) and materials properties are downloaded from the Materials Project database (V2021.03.22), and those for κ are from the TEDesignLab database. U and C_v are calculated by the PhononDos class in pymatgen based on the phonon density of states from the Materials Project database. The dataset size for each property, shown in Table S1 in the Supplementary Information, depends on the number of structures that have the recorded property in the two materials databases. For machine learning of materials properties in Figure 4, we split the datasets into 60%, 20% and 20% as the training, validation and test set.

For the dataset used for testing whether CGCNN and ALIGNN can capture human-designed descriptors of crystal structures, since we know that lattice constants of high-symmetry materials are reported to be more learnable than that of low-symmetry materials based on compositions, we create a subset of the Materials Project database (“MP dataset”) by removing some structures randomly based on their space group number:

\[
\text{Probability(removed)} = \frac{\text{Space group number}}{\text{Space group number} + 15},
\]

where 15 is the space group number of the C2/c group, the last space group in the class of monoclinic Bravais lattice. Consequently, we have a dataset with 47,862 crystal structures biased to materials with low symmetry to test whether CGCNN and ALIGNN can learn human-designed descriptors from crystal structures. To facilitate the analysis about failure of CGCNN to capture lattice constants, we create a dataset of random 1-dimensional carbon chains (“1D
dataset”). The random 1D chains are created by the following pseudo-codes in python:

```python
pos = []; for j in range(n):  # number of atoms in the chain
    if j == 0: pos.append([3*random for 3 dimensions])  # 3 = 2*1.5 Å (approx. C-C bond length).
    # random: random number between (0, 1)
    elif j%2 == 0: pos.append([pos[j-1] + 3*random for 3 dimensions])
    else: pos.append([pos[j-1][0] + 3*random, pos[j-1][1] - 3*random, pos[j-1][2] - 3*random])

a = pos[-1][0]; b = 100; c = 100  # add vacuum for b and c
lattice = Lattice.from_parameters(a, b, c, 90, 90, 90)
structure = pymatgen.core.structure.Structure(lattice, "C" for _ in range(n)], pos, coords_are_cartesian=True)
```

For the dataset of (1D, short), the number of atoms is set to be between [2, 9), and for the dataset of (1D, long), the number of atoms is set to be between [37, 51). In total, both datasets have 1,400 data points. For machine learning of human-designed descriptors in Figure 2 and 3, we split the dataset into 80%, 10% and 10% as the training, validation and test set.

Table 1. List of abbreviations of descriptors and properties in this work.

| Abbreviations of descriptors | Full name of descriptors | Abbreviations of properties | Full name of descriptors |
|-----------------------------|--------------------------|----------------------------|--------------------------|
| MAD_in_rela_bond_len        | mean absolute deviation in relative bond length | log(κ)                     | log10 lattice thermal conductivity |
| max_rela_bond_len           | maximum relative bond length | $E_{\text{fin.}}$          | final (total) energy per atom |
| min_rela_bond_len           | minimum relative bond length | $U$                        | phonon internal energy at 300 K |
| max_neigh_dist_var          | maximum neighbor distance variation | $C_v$                      | constant volume phonon heat capacity at 300 K |
| **min_neighb_dist_var** | minimum neighbor distance variation | $K$ | bulk modulus |
| **range_neighb_dist_var** | range neighbor distance variation | $G$ | shear modulus |
| **mean_neighb_dist_var** | mean neighbor distance variation | $v$ | poisson ratio |
| **dev_neighb_dist_var** | standard deviation neighbor distance variation | $E_g$ | band gap |
| **mean_avg_bond_len** | mean average bond length | $\|\varepsilon\|_\infty$ | modulus of piezoelectric tensor |
| **std_avg_bond_len** | standard deviation average bond length | $\varepsilon_e$, $\varepsilon_t$ | electronic dielectric constant |
| **MAD_in_rela_atom_vol** | mean absolute deviation in atomic volume | $\varepsilon_e$, $\varepsilon_t$ | total dielectric constant |
| **mean_avg_bond_ang** | mean average bond angle | $n$ | refractive index |
| **std_avg_bond_ang** | standard deviation average bond angle | $M$ | total magnetization per formula |
| density | density |
| **vpa** | volume per atom |
| **packing_frac** | packing fraction |
| **struct_comp_atom** | structural complexity per atom |
| **struct_comp_cell** | structural complexity per primitive cell |
| **natoms** | number of atoms per primitive cell |
| $a$ | the largest lattice length of the primitive cell |
| $b$ | the second largest lattice length of the primitive cell |
| $c$ | the smallest lattice length of the primitive cell |
| $\alpha$ | the largest lattice angle of the primitive cell |
| $\beta$ | the second largest lattice angle of the primitive cell |
Models. In this work, we use the default architecture of CGCNN\(^3\) and ALIGNN\(^2\) for learning human-designed descriptors in Figure 2 and 3 unless specifically mentioned. The reason for using the default architectures is that, as in Figure 3g and 3h, although intentionally revising their architectures can improve learning performance for some descriptors, in this work we try to show the representational power and limit of CGCNN and ALIGNN in a setting close to those in real applications. For learning materials properties in Figure 4, hyper-parameter search based on the search spaces in Table S2 and S3 is conducted; All the neural networks are trained for 300 epochs\(^2\) on a Quadro RTX 6000 GPU. For feature importance in Figure 4, since the permutation feature importance of deep neural networks is very expensive to calculate, we estimate the feature importance by extracting the representations in equation (12), then feed the representations into a random forest model to calculate the feature importance.

Data & code availability
All datasets, codes and trained machine learning models in this work are provided at:
https://figshare.com/articles/journal_contribution/Improving_deep_representation_learning_for_crystal_structures_by_learning_and_hybridizing_human-designed_descriptors/19654224

References
1. Kong, S.; Ricci, F.; Guevarra, D.; Neaton, J. B.; Gomes, C. P.; Gregoire, J. M., Density of states prediction for materials discovery via contrastive learning from probabilistic embeddings. Nat Commun 2022, 13 (1), 949.
2. Mao, J.; Chen, G.; Ren, Z. J. N. M., Thermoelectric cooling materials. Nat. Mater. 2021, 20 (4), 454-461.
3. Cheng, J.; Zhang, C.; Dong, L., A geometric-information-enhanced crystal graph network for predicting properties of materials. Commun. Mater. 2021, 2 (1).
4. Karamad, M.; Magar, R.; Shi, Y.; Siahrostami, S.; Gates, I. D.; Barati Farimani, A., Orbital graph convolutional neural network for material property prediction. Phys. Rev. Mater. 2020, 4 (9).
5. Ahmad, Z.; Xie, T.; Maheshwari, C.; Grossman, J. C.; Viswanathan, V., Machine Learning Enabled Computational Screening of Inorganic Solid Electrolytes for Suppression of Dendrite Formation in Lithium Metal Anodes. ACS Cent Sci 2018, 4 (8), 996-1006.
6. Yamada, H.; Liu, C.; Wu, S.; Koyama, Y.; Ju, S.; Shiomi, J.; Morikawa, J.; Yoshida, R., Predicting Materials Properties with Little Data Using Shotgun Transfer Learning. ACS Cent Sci 2019, 5 (10), 1717-1730.
7. Janet, J. P.; Kulik, H. J., Predicting electronic structure properties of transition metal complexes with neural networks. Chem Sci 2017, 8 (7), 5137-5152.
8. Zheng, X.; Zheng, P.; Zhang, R. Z., Machine learning material properties from the periodic table using convolutional neural networks. Chem Sci 2018, 9 (44), 8426-8432.
9. Patra, A.; Batra, R.; Chandrasekaran, A.; Kim, C.; Huan, T. D.; Ramprasad, R., A multi-fidelity information-fusion approach to machine learn and predict polymer bandgap. Comput. Mater. Sci. 2020, 172, 109286.
10. Pilania, G.; Gubernatis, J. E.; Lookman, T., Multi-fidelity machine learning models for accurate bandgap predictions of solids. Comput. Mater. Sci. 2017, 129, 156-163.
11. Chen, L.; Tran, H.; Batra, R.; Kim, C.; Ramprasad, R., Machine learning models for the lattice thermal conductivity prediction of inorganic materials. Comput. Mater. Sci. 2019, 170.
12. Zhu, T.; He, R.; Gong, S.; Xie, T.; Gorai, P.; Nielsch, K.; Grossman, J. C., Charting lattice thermal conductivity for inorganic crystals and discovering rare earth chalcogenides for thermoelectrics. Energy Environ. Sci. 2021, 14 (6), 3559-3566.
13. Jang, J.; Gu, G. H.; Noh, J.; Kim, J.; Jung, Y., Structure-Based Synthesizability Prediction of Crystals Using Partially Supervised Learning. J Am Chem Soc 2020, 142 (44), 18836-18843.
14. Schutt, K. T.; Sauceda, H. E.; Kindermans, P. J.; Tkatchenko, A.; Muller, K. R., SchNet - A deep learning architecture for molecules and materials. J Chem Phys 2018, 148 (24), 241722.
15. Back, S.; Yoon, J.; Tian, N.; Zhong, W.; Tran, K.; Ulissi, Z. W., Convolutional Neural Network of Atomic Surface Structures To Predict Binding Energies for High-Throughput Screening of Catalysts. J Phys Chem Lett 2019, 10 (15), 4401-4408.
16. Liang, J.; Zhu, X., Phillips-Inspired Machine Learning for Band Gap and Exciton Binding Energy Prediction. J Phys Chem Lett 2019, 10 (18), 5640-5646.
17. Gong, S.; Wang, S.; Zhu, T.; Chen, X.; Yang, Z.; Buehler, M. J.; Shao–Horn, Y.; Grossman, J. C., Screening and Understanding Li Adsorption on Two-Dimensional Metallic Materials by Learning Physics- and Physics-Simplified Learning. JACS Au 2021.
18. Gupta, V.; Choudhary, K.; Tavazza, F.; Campbell, C.; Liao, W. K.; Choudhary, A.; Agrawal, A., Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data. Nat Commun 2021, 12 (1), 6595.
19. Isayev, O.; Oses, C.; Toher, C.; Gossett, E.; Curtarolo, S.; Tropsha, A., Universal fragment descriptors for predicting properties of inorganic crystals. *Nat Commun* 2017, 8, 15679.

20. Banjade, H. R.; Hauri, S.; Zhang, S.; Ricci, F.; Gong, W.; Hautier, G.; Vucetic, S.; Yan, Q., Structure motif centric learning framework for inorganic crystalline systems. *Sci. Adv.* 2021, 7(17), eabf1754.

21. Jha, D.; Choudhary, K.; Tavazza, F.; Liao, W. K.; Choudhary, A.; Campbell, C.; Agrawal, A., Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning. *Nat Commun* 2019, 10(1), 5316.

22. Ye, W.; Chen, C.; Wang, Z.; Chu, I. H.; Ong, S. P., Deep neural networks for accurate predictions of crystal stability. *Nat Commun* 2018, 9(1), 3800.

23. Chen, C.; Zuo, Y.; Ye, W.; Li, X.; Ong, S. P., Learning properties of ordered and disordered materials from multi-fidelity data. *Nat. Comput. Sci.* 2021, 1(1), 46-53.

24. Bartel, C. J.; Trewartha, A.; Wang, Q.; Dunn, A.; Jain, A.; Ceder, G., A critical examination of compound stability predictions from machine-learned formation energies. *npj Comput. Mater.* 2020, 6(1).

25. Kailkhura, B.; Gallagher, B.; Kim, S.; Hiszpanski, A.; Han, T. Y.-J., Reliable and explainable machine-learning methods for accelerated material discovery. *Npj Comput. Mater.* 2019, 5(1).

26. Ward, L.; Agrawal, A.; Choudhary, A.; Wolverton, C., A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Comput. Mater.* 2016, 2(1).

27. Choudhary, K.; DeCost, B., Atomistic Line Graph Neural Network for improved materials property predictions. *npj Comput. Mater.* 2021, 7(1).

28. Ghiringhelli, L. M.; Vybiral, J.; Levchenko, S. V.; Draxl, C.; Scheffler, M., Big data of materials science: critical role of the descriptor. *Phys Rev Lett* 2015, 114(10), 105503.

29. Xie, T.; Grossman, J. C., Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties. *Phys Rev Lett* 2018, 120(14), 145301.

30. Choudhary, K.; DeCost, B.; Tavazza, F., Machine learning with force-field inspired descriptors for materials: fast screening and mapping energy landscape. *Phys Rev Mater* 2018, 2(8).

31. Gong, S.; Xie, T.; Zhu, T.; Wang, S.; Fadel, E. R.; Li, Y.; Grossman, J. C., Predicting charge density distribution of materials using a local-environment-based graph convolutional network. *Phys. Rev. B* 2019, 100(18), 184103.

32. Schütt, K. T.; Glawe, H.; Brockherde, F.; Sanna, A.; Müller, K. R.; Gross, E. K. U., How to represent crystal structures for machine learning: Towards fast prediction of electronic properties. *Phys. Rev. B* 2014, 89(20), 205118.

33. Park, C. W.; Wolverton, C., Developing an improved crystal graph convolutional neural network framework for accelerated materials discovery. *Phys. Rev. Mater.* 2020, 4(6), 063801.

34. Batra, R.; Song, L.; Ramprasad, R., Emerging materials intelligence ecosystems propelled by machine learning. *Nat. Rev. Mater.* 2020, 6(8), 655-678.

35. Faber, F.; Lindmaa, A.; von Lilienfeld, O. A.; Armiento, R., Crystal structure representations for machine learning models of formation energies. *Int. J. Quantum Chem.* 2015, 115(16), 1094-1101.

36. Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P., Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chem. Mater.* 2019, 31(9), 3564-3572.

37. Louis, S. Y.; Zhao, Y.; Nasiri, A.; Wang, X.; Song, Y.; Liu, F.; Hu, J., Graph convolutional neural networks with global attention for improved materials property prediction. *Phys Chem Chem Phys* 2020, 22(32), 18141-18148.
38. Gasteiger, J.; Groß, J.; Günnemann, S., DIRECTIONAL MESSAGE PASSING FOR MOLECULAR GRAPHS. *International Conference on Learning Representations* 2020.
39. Gasteiger, J.; Becker, F.; Günnemann, S., GemNet: Universal Directional Graph Neural Networks for Molecules. In *35th Conference on Neural Information Processing Systems* 2021.
40. Maron, H.; Litany, O.; Chechik, G.; Fetaya, E., On Learning Sets of Symmetric Elements. In *Proceedings of the 37th International Conference on Machine Learning*, Hal, D., III; Aarti, S., Eds. PMLR: Proceedings of Machine Learning Research, 2020; Vol. 119, pp 6734--6744.
41. Morris, C.; Rattan, G.; Mutze, P., Weisfeiler and Leman go sparse: Towards scalable higher-order graph embeddings. *34th Conference on Neural Information Processing Systems* 2020.
42. Hsu, T.; Pham, T. A.; Keilbart, N.; Weitzner, S.; Chapman, J.; Xiao, P.; Qiu, S. R.; Chen, X.; Wood, B. C., Efficient and Interpretable Graph Neural Network Representation for Angle-dependent Properties and its Application to Optical Spectroscopy. *npj Comput Mater* 2022, 8, 151.
43. Batatia, I.; Batzner, S.; Kovacs, D. P.; Musaelian, A.; Simm, G. N. C.; Draut, R.; Ortner, C.; Kozinsky, B.; Csanýi, G., The Design Space of E(3)-equivariant Atom-Centered Interatomic Potentials. arXiv:2205.06643v1 2022.
44. Batzner, S.; Musaelian, A.; Sun, L.; Geiger, M.; Mailoa, J. P.; Kornbluth, M.; Molinari, N.; Smidt, T. E.; Kozinsky, B., E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat Commun* 2022, 13(1), 2453.
45. Zhong, Y.; Yu, H.; Gong, X.; Xiang, H., Edge-based Tensor prediction via graph neural networks. arXiv:2201.05770 2022.
46. Dunn, A.; Wang, Q.; Ganose, A.; Dopp, D.; Jain, A., Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Comput. Mater.* 2020, 6(1), 138.
47. A. Jain*, S. P. O., G. Hautier, W. Chen, W.D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, K.A. Persson, The Materials Project: A materials genome approach to accelerating materials innovation. *APL Mater.* 2013, 1(1), 011002.
48. Goodall, R. E. A.; Lee, A. A., Predicting materials properties without crystal structure: deep representation learning from stoichiometry. *Nat Commun* 2020, 11(1), 6280.
49. Ward, L.; Dunn, A.; Faghaninia, A.; Zimmermann, N. E. R.; Bajaj, S.; Wang, Q.; Montoya, J.; Chen, J.; Bystrom, K.; Dylla, M.; Chard, K.; Asta, M.; Persson, K. A.; Snyder, G. J.; Foster, I.; Jain, A., Matminer: An open source toolkit for materials data mining. *Comput. Mater. Sci.* 2018, 152, 60–69.
50. Ong, S. P.; Richards, W. D.; Jain, A.; Hautier, G.; Kocher, M.; Cholia, S.; Gunter, D.; Chevrier, V. L.; Persson, K. A.; Ceder, G., Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Comput. Mater. Sci.* 2013, 68, 314–319.
51. Grosse-Kunstleve, R. W.; Sauter, N. K.; Adams, P. D., Numerically stable algorithms for the computation of reduced unit cells. *Acta Crystallogr. A* 2004, 60(Pr 1), 1-6.
52. WOLFF, P. M. D.; GRUBER, B., Niggli Lattice Characters. *Acta Cryst.* 1991, A47, 29–36.
53. Sabirov, D. S.; Shepelevich, I. S., Information Entropy in Chemistry: An Overview. *Entropy (Basel)* 2021, 23(10).
54. Li, Y.; Yang, W.; Dong, R.; Hu, J., Mlatticeabc: Generic Lattice Constant Prediction of Crystal Materials Using Machine Learning. *ACS Omega* 2021, 6(17), 11585–11594.
55. Liang, H.; Stanjev, V.; Kusne, A. G.; Takeuchi, I., CRYSPNet: Crystal structure predictions via neural networks. *Phys. Rev. Mater.* 2020, 4(12).
56. Chen, Z.; Villar, S.; Chen, L.; Bruna, J., On the equivalence between graph isomorphism testing and function approximation with GNNs. *33rd Conference on Neural Information Processing Systems*. 2019, 15868.

57. Saelim, T.; Chainok, K.; Kielar, F.; Wannarit, N., Crystal structure of a novel one-dimensional zigzag chain-like cobalt(II) coordination polymer constructed from 4,4′-bi-pyridine and 2-hy-droxy-benzoate ligands. *Acta Crystallogr E Crystallogr Commun* 2020, *76*(Pt 8), 1302-1306.

58. Sarkar, A. S.; Stratakis, E., Recent Advances in 2D Metal Monochalcogenides. *Adv. Sci.* 2020, 7(21), 2001655.

59. Alon, U.; Yahav, E., ON THE BOTTLENECK OF GRAPH NEURAL NETWORKS AND ITS PRACTICAL IMPLICATIONS. *International Conference on Learning Representations*. 2021.

60. Li; Q.; Han; Z.; Xiao-MingWu, Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning. *AAAI Conference on Artificial Intelligence*. 2018, 3538.

61. Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S., HOW POWERFUL ARE GRAPH NEURAL NETWORKS. *International Conference on Learning Representations*. 2019.

62. Omee, S. S.; Louis, S.-Y.; Fu, N.; Lai Wei; Dey, S.; Dong, R.; Li, Q.; Hu, J., Scalable deeper graph neural networks for high-performance materials property prediction. *Patterns* 2022, 3(5), 100491.

63. Bodnar, C.; Frasca, F.; Wang, Y. G.; Otter, N.; Mont’ufar, G.; Li, P.; Bronstein, M. M., Weisfeiler and Lehman Go Topological: Message Passing Simplicial Networks. *Proceedings of the 38th International Conference on Machine Learning*. 2021, 139.

64. Bodnar, C.; Frasca, F.; Otter, N.; Wang, Y. G.; Liò, P.; Montúfar, G.; Bronstein, M., Weisfeiler and Lehman Go Cellular: CW Networks. *35th Conference on Neural Information Processing Systems*. 2021.

65. Gu, F.; Chang, H.; Zhu, W.; Sojoudi, S.; Ghaoui, L. E., Implicit Graph Neural Networks. *34th Conference on Neural Information Processing Systems*. 2020.

66. ZhanghaoWu; Jain, P.; Wright, M. A.; Mirhoseini, A.; Gonzalez, J. E.; Stoica, I., Representing Long-Range Context for Graph Neural Networks with Global Attention. *35th Conference on Neural Information Processing Systems*. 2021.

67. <b-doped-graphite-li-anode.pdf>.

68. Kittel, C., Introduction to solid state physics Eighth edition. 2021.

69. Legrain, F.; Carrete, J.; van Roekeghem, A.; Curtarolo, S.; Mingo, N., How Chemical Composition Alone Can Predict Vibrational Free Energies and Entropies of Solids. *Chem. Mater.* 2017, 29(15), 6220-6227.

70. Yan, J.; Gorai, P.; Ortiz, B.; Miller, S.; Barnett, S. A.; Mason, T.; Stevanović, V.; Toberer, E. S., Material descriptors for predicting thermoelectric performance. *Energy Environ. Sci.* 2015, 8(3), 983-994.

71. Gorai, P.; Gao, D.; Ortiz, B.; Miller, S.; Barnett, S. A.; Mason, T.; Lv, Q.; Stevanović, V.; Toberer, E. S., TE Design Lab: A virtual laboratory for thermoelectric material design. *Comput. Mater. Sci.* 2016, 112, 368-376.

72. Kirklin, S.; Saal, J. E.; Meredig, B.; Thompson, A.; Doak, J. W.; Aykol, M.; Rühl, S.; Wolverton, C., The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies. *npj Comput. Mater.* 2015, 1(1).

73. Curtarolo, S.; Setyawan, W.; Wang, S.; Xue, J.; Yang, K.; Taylor, R. H.; Nelson, L. J.; Hart, G. L. W.; Sanvito, S.; Buongiorno-Nardelli, M.; Mingo, N.; Levy, O., AFLOWLIB.ORG: A distributed materials properties repository from high-throughput ab initio calculations. *Comput. Mater. Sci.*
Choudhary, K.; Garrity, K. F.; Reid, A. C.; DeCost, B.; Biacchi, A. J.; Walker, A. R. H.; Trautt, Z.; Hattrick-Simpers, J.; Kusne, A. G.; Centrone, A. J. a. p. a., JARVIS: An Integrated Infrastructure for Data-driven Materials Design. \textit{arXiv:2007.01831} 2020.

Ortiz-Jiménez, G.; Modas, A.; Moosavi-Dezfooli, S.-M.; Frossard, P., Hold me tight! Influence of discriminative features on deep network boundaries. \textit{34th Conference on Neural Information Processing Systems} 2020.

Cai, J.; Luo, J.; Wang, S.; Yang, S., Feature selection in machine learning: A new perspective. \textit{Neurocomputing} 2018, 300, 70-79.
Supplementary Information for:

Examining graph neural networks for inorganic crystalline structures: limitation of capturing periodicity

Sheng Gong¹, Tian Xie²,³, Yang Shao-Horn¹,⁴, Rafael Gomez-Bombarelli¹, and Jeffrey C. Grossman¹,*

¹Department of Materials Science and Engineering, Massachusetts Institute of Technology, MA 02139, USA
²Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology, MA 02139, USA
³Microsoft Research, Cambridge, United Kingdom CB1 2FB
⁴Department of Mechanical Engineering, Massachusetts Institute of Technology, MA 02139, USA

Contents
1. Limitation of average pooling on extensive properties.................................................................36
2. Limitation of sum pooling on capturing periodicity.............................................................38
3. Learning performance and artificial choice of lattice angles.........................................................39
4. Structural Complexity.................................................................................................................40
5. More discussions about $U$ and $C_v$ .........................................................................................41
6. Impact of hybridization with descriptors on the learned representations.....................................41
7. Supplementary Tables.................................................................................................................45
1. Limitation of average pooling on extensive properties.

Suggested by the limitation of average pooling on capturing primitive cell-level information, in this section, we show the limitation of average pooling on prediction of extensive properties. Extensive properties scale with number of atoms, such as total energy ($E_{\text{fin.}}$), (phonon) internal energy ($U$), (phonon) heat capacity ($C_v$) and magnetization ($M$). Despite their extensive nature, in many cases these extensive properties are normalized to intensive versions for data storage and evaluation \textsuperscript{27, 29, 36, 46, 47, 72-74}, such as energy per atom and magnetization per unit formula. Since intensive properties cannot be learned by sum pooling because of their size invariance\textsuperscript{29}, for convenience, many GNNs, such as CGCNN and ALIGNN, only implement average pooling and learn the intensive versions of these extensive properties by average pooling in equation (4). Another way to learn extensive properties is illustrated in the bottom branch in Figure S1a, where extensive properties are first learned by machine learning models with sum pooling in equation (13), then normalized for evaluation.

Do the two approaches for learning extensive properties in Figure S1a have the same predictive power? Intuitively, they seem to be equivalent. However, as suggested in Figure 3c and 3h, we know that sum pooling is more powerful to distinguish graphs, and sum pooling can capture periodicity better. This insight suggests that sum pooling might be more powerful than average pooling for learning extensive properties. As shown in Figure S1b, we show the MAE/MAD scores of predictions of four extensive properties by CGCNN and ALIGNN with average pooling and sum pooling. We can see that, sum pooling outperforms average pooling for $U$, $C_v$ and $M$, and has similar performance for $E_{\text{fin.}}$, which verifies our hypothesis that sum pooling might be more powerful than average pooling for learning extensive properties.

Currently, many GNNs designed for prediction of materials properties, such as CGCNN and ALIGNN, only implement size-invariant average pooling function, with the exception of
MEGNet where users can easily switch between sum pooling and average pooling. Based on the results shown in this section, we argue that GNN models designed for prediction of materials properties should provide options of pooling functions for different properties, such as average pooling for intensive properties and sum pooling for extensive properties.

Figure S1. Average pooling and sum pooling for extensive properties. a Schematic of two approaches to learn, predict and evaluate prediction performance of extensive properties. b MAE/MAD of prediction of four extensive properties by CGCNN and ALIGNNN with average pooling and sum pooling.
2. Limitation of sum pooling on capturing periodicity.

As mentioned in the main text, even if only primitive cells are input to the GNNs, sum pooling might also fail to capture periodicity in some cases, as periodicity does not always scale with number of atoms in the primitive cells. For example, in Figure 3d we show the case of 1D double chains. Compared with 1D single chains in Figure 3b and 3c, 1D double chains can have similar periodicity but twice number of atoms. In Figure S2, we show the $R^2$ scores of predictions of $a$ of the datasets with 1D single short chains, 1D single and double short chains, 1D single long chains, and 1D single and double long chains, from CGCNN with average pooling and sum pooling, respectively. For the datasets with only single chains, sum pooling outperforms average pooling significantly, while for the datasets with single and double chains, sum pooling can only have very limited improvement over average pooling, which shows the co-existence of single and double chains makes sum pooling harder to determine periodicity than the case with only single chains.

**Figure S2.** $R^2$ scores of predictions of $a$ of the datasets with 1D single short chains, 1D single and double short chains, 1D single long chains, and 1D single and double long chains, from
CGCNN with average pooling and sum pooling, respectively.

3. Learning performance and artificial choice of lattice angles.

As mentioned in the main text, both CGCNN and ALIGNN cannot learn the lattice angles well. In Figure S3a, we show the prediction performance of CGCNN, CGCNN with more convolutions, CGCNN with larger limits of number of neighbors and neighboring cut-off radius, CGCNN with sum pooling and ALIGNN for three lattice angles. We can see that both CGCNN and ALIGNN cannot learn the three lattice angles well, and the modifications that improve learning performance of lattice lengths as in the main text do not improve that of lattice angles.

The artificial choice of lattice angles might cause the poor learning performance. As stated in the main text, we use the Niggli reduction to determine the unique Niggli-primitive cell for any given periodic structure. In crystallography, we can use a set of six parameters to define a primitive cell: \((a, b, c, \alpha, \beta, \gamma)\), where the first three parameters are lengths of lattice vectors, and the later three parameters are angles between lattice vectors. In Niggli reduction, we require the lengths of lattice vectors to satisfy the following requirement:

\[ a + b + c = \text{minimum} \quad \text{...(S1)} \]

If the order of the three lengths are defined, as in the main text we define \(a\) the largest one and \(c\) the smallest one, then the set of \((a, b, c)\) is unique for any given structure. However, the set of \((\alpha, \beta, \gamma)\) might not be unique even if \((a, b, c)\) is unique. For example, in Figure S3b, we plot the structure of simple hexagonal structure. We can see that the three lengths of lattice vectors are unique and reflect the intrinsic characteristics of the structure, such as the three minimal distances between the smallest repeating units in three dimensions. For \((\alpha, \beta, \gamma)\), we know that
two of them are 90° (angles between the vertical lattice vector and the two in-plane lattice vectors), but there are actually two choices of the third one: 60° and 120°, and two choices of $(\alpha, \beta, \gamma)$: $(120°, 90°, 90°)$ and $(90°, 90°, 60°)$. Although in Niggli reduction, $(120°, 90°, 90°)$ is chosen as the lattice angles, this choice is artificial to the given structures. Therefore, machine learning algorithms might not be able to capture the artificial characteristics of crystal structures. Further studies are necessary to design more intrinsic description of relative orientation between the lattice vectors.

Figure S3. a R² scores of prediction of lattice angles from CGCNN, ALIGNN and variants of CGCNN. Alpha denotes the largest lattice angle, and gamma the smallest one. b Illustrations of ambiguity of choice of lattice angles by the example of simple hexagonal primitive cell.

4. Structural Complexity

Structural complexity is defined as below:

$$S = -N \sum_{i=1}^{k} \frac{m_i}{N} \log_2 \frac{m_i}{N} ... (S2),$$

where $N$ is the total number of atoms in the primitive cell, $k$ is the number of symmetrically inequivalent sites, $m_i$ is the number of sites classified in the $i^{th}$ symmetrically inequivalent site.
Structural complexity quantifies the complexity of sites distribution in a structure, as larger complexity, more different symmetrically inequivalent sites in the structure. Structural complexity per primitive cell is calculated as equation S2, and structural complexity per atom is calculated by equation S2 divided by the number of atoms in the primitive cell.

5. More discussions about $U$ and $C_V$

Recently, Legrain et al.\textsuperscript{69} reported machine learning of $U$ by compositions of materials. The main difference between this work and Legrain et al.\textsuperscript{69} is that, in Legrain et al.\textsuperscript{69}, only 292 materials are included in the dataset, while in this work about 1,500 materials are included in the datasets for $U$ and $C_V$. On the other hand, with the same dataset as this work, Mat2Spec\textsuperscript{1} and E3NN\textsuperscript{2} are proposed to predict phonon density of states, and consequently, heat capacity. Unfortunately, we cannot easily compare our predictions and predictions from the two models mentioned above, as the $C_V$ in this work is based on full phonon density of states, while Mat2Spec\textsuperscript{1} and E3NN\textsuperscript{2} are designed to predict filtered and truncated phonon density of states with 51 frequencies up to 1000 cm$^{-1}$. The reason for the success of E3NN for predicting $C_V$ might be that E3NN employs equivariant representations\textsuperscript{2} with high local expressive power, and the reason for the success of Mat2Spec for predicting $C_V$ might be the explicit exploitation of correlations of density of states between frequencies in phonon density of states.

6. Impact of hybridization with descriptors on the learned representations

As in equation (14) in the main text, since the descriptors participate in the optimization of deep representation learning, the inclusion of descriptors would affect the optimization of the
learned representations. In other words, consider the gradient propagation in the optimization of the representation learning:

$$\frac{\partial L}{\partial w_{pq}^R} = \frac{\partial z_{q}^{R+1}}{\partial w_{pq}^R} * \frac{\partial L}{\partial z_{q}^{R+1}} ...... (S3),$$

where $L$ is the loss function, $w_{pq}^R$ is the weight from the $p^{th}$ unit of the representation layer \((\frac{1}{N_a} \sum_{i=1}^{N_a} a_i^n \oplus \text{descriptors})\) in equation (14) to the $q^{th}$ unit of the layer after the representation layer (first layer of FCN in equation (14)), and $z_{q}^{R+1}$ is the $q^{th}$ unit of the layer after the representation layer. Therefore, after inclusion of descriptors, $z_{q}^{R+1}$ changes, and consequently $\frac{\partial L}{\partial w_{pq}^R}$ changes even if the $p^{th}$ unit is from the part $\frac{1}{N_a} \sum_{i=1}^{N_a} a_i^n$, and all gradients before the representation layer change due to the chain rule of gradient propagation.

How does the change of gradient affect the learned representations? As an example, in Figure 4c, we show the feature importance spectrum of de-CGCNN for prediction of $\kappa$, from which we can see that the learned representations play the most important roles, and some descriptors contribute significantly such as mean and standard deviation of bond length, volume per atom and $b$. Except $b$, the other three important descriptors can be well learned by CGCNN as shown in Figure 2a. In order to understand this phenomenon, we investigate how well the learned representations from CGCNN correlate with descriptors before and after the hybridization with descriptors. In other words, we investigate the following correlation:

$$\text{Corr}( \frac{1}{N_a} \sum_{i=1}^{N_a} a_i^n, \text{descriptors}) ......(S4).$$

In Figure S4a, we show the correlation between the learned representations and each descriptor for learning $\kappa$, and we can see that the learned representations from de-CGCNN are less correlated with the descriptors than that from CGCNN. The weaker correlation after the inclusion of descriptors supports our hypothesis that hybridization with descriptors pushes the
optimization of learned representations away from the already known information in the input human-designed descriptors.

As a comparison, we construct machine learning models based on learned representations from CGCNN and human-designed descriptors as below:

\[
\text{Output} = \text{FCN}(\frac{1}{N_a} \sum_{i=1}^{N_a} a_i^n \ast (\text{already learned from CGCNN}) \oplus \text{descriptors}) \quad \ldots \quad (S5).
\]

The main difference between equation (S5) (named as “CGCNN+descriptors”) and equation (14) (de-CGCNN) is that, descriptors participate in the optimization of learned representations in de-CGCNN, while descriptors do not in CGCNN+descriptors. In Figure S4b, we show the MAE/MAD ratio of machine learning models based only on descriptors, CGCNN, CGCNN+descriptors, and de-CGCNN. We can see that, for most properties, de-CGCNN has lower error than CGCNN+descriptors, which supports the proposed mechanism that participation of descriptors in the optimization of representations improves the performance of deep representation learning. For \( \kappa \), \( U \), \( C_v \), and \( M \), we observe that CGCNN+descriptors outperforms CGCNN, and such improvement mainly comes from the missing information in descriptors as discussed in the main text. For most remaining properties, CGCNN+descriptors has similar performance with CGCNN, which suggests that the improvement of de-CGCNN for these properties might come from the participation of descriptors in the optimization of representations.

Despite the observed improvement, intuitively, hybridization with descriptors in equation (14) and addition of descriptors in equation (S5) would have a negative impact on prediction performance due to the introduction of redundant information from these two modifications\textsuperscript{75,76}. Such redundancy can be seen in Figure S4a, where learned representations from both CGCNN and de-CGCNN are correlated with human-designed descriptors in some degree. In Figure S4b, we observe that CGCNN+descriptors has higher error than the model based on
only descriptors for $U$ and $C_v$. Since the only difference between the two models is the presence of learned representations in CGCNN+descriptors, the increase of error associated with CGCNN+descriptors supports the proposed mechanism that redundant information can harm the predictive power of machine learning models\textsuperscript{75, 76}.

Figure S4. a $R^2$ scores of linear regressions between each descriptor and the learned representations from CGCNN and de-CGCNN. b MAE/MAD ratio of prediction of 13
materials properties from machine learning models based only on descriptors, CGCNN, machine learning models based on CGCNN-learned representations and descriptors, and de-CGCNN.

### 7. Supplementary Tables

Table S1. Ratio of materials (related to the 139,367 materials in the Materials Project V2021.03.22) with each property and prediction results of machine learning models for the lattice thermal conductivity in the TEDesignLab database and 12 properties in the Materials Project database.

|        | Unit   | Ratio of | MAD  | MAE of | MAE of | MAE of | MAE of | MAE of |
|--------|--------|----------|------|--------|--------|--------|--------|--------|
|        |        | materials|      | descriptors + RF | CGCNN + descriptors | de-CGCNN | CGCNN | ALIGNN |
| log (κ) |        | 0.020    | 0.427| 0.156  | 0.137  | 0.111  | 0.148  | 0.117  | 0.110  |
|        |        |          |      | (0.365) | (0.320) | (0.260) | (0.347) | (0.276) | (0.248) |
| E\(_{\text{fim.}}\) | eV/atom | 1.0      | 1.289| 0.513  | 0.071  | 0.054  | 0.073  | 0.058  | 0.068  |
|        |        |          |      | (0.398) | (0.055) | (0.042) | (0.057) | (0.045) | (0.052) |
| U      | KJ/mol- | 0.011    | 25.70| 3.602  | 4.806  | 1.340  | 17.22  | 13.73  | 2.617  |
|        | e  atom |          |      | (0.140) | (0.187) | (0.060) | (0.670) | (0.534) | (0.101) |
| U      | KJ/mol- | 0.011    | 5.106| 3.047  | 2.996  |        |        |        |        |
|        | atom    |          |      | (0.602) | (0.587) |        |        |        |        |
| C\(_v\) | J/(mol- | 0.011    | 55.61| 6.156  | 8.286  | 3.249  | 42.19  | 35.18  | 4.952  |
|        | e*K)    |          |      | (0.111) | (0.149) | (0.058) | (0.759) | (0.632) | (0.089) |
| C\(_v\) | J/(mol- | 0.011    | 13.81| 8.301  | 8.374  |        |        |        |        |
|        | atom*K) |          |      | (0.601) | (0.606) |        |        |        |        |
| K      | GPa    | 0.095    | 56.41| 19.62  | 12.15  | 12.15  | 12.26  | 11.49  | 11.61  |
|        |        |          |      | (0.348) | (0.215) | (0.215) | (0.217) | (0.204) | (0.206) |
| G      | GPa    | 0.095    | 30.63| 14.98  | 12.01  | 11.08  | 12.17  | 10.31  | 9.988  |
|        |        |          |      | (0.489) | (0.392) | (0.362) | (0.397) | (0.337) | (0.326) |
| ν      |        | 0.095    | 0.064| 0.051  | 0.050  | 0.050  | 0.084  | 0.046  | 0.045  |
|        |        |          |      | (0.800) | (0.788) | (0.788) | (0.801) | (0.716) | (0.702) |
| E\(_g\) | eV    | 1.0      | 1.258| 0.643  | 0.282  | 0.274  | 0.290  | 0.247  | 0.249  |
|        |        |          |      | (0.511) | (0.224) | (0.218) | (0.231) | (0.196) | (0.198) |
| ||e||\(_c\) | C/m\(^2\) | 0.024 | 0.555| 0.440  | 0.452  | 0.423  | 0.451  | 0.442  | 0.411  |
|        |        |          |      | (0.793) | (0.815) | (0.762) | (0.813) | (0.796) | (0.742) |
| ε\(_e\) |        | 0.052    | 2.819| 1.399  | 0.736  | 0.683  | 0.744  | 0.602  | 0.604  |
|        |        |          |      | (0.496) | (0.261) | (0.242) | (0.264) | (0.213) | (0.214) |
| ε\(_t\) |        | 0.052    | 9.055| 6.064  | 5.125  | 4.971  | 5.102  | 4.507  | 4.432  |
|        |        |          |      | (0.670) | (0.566) | (0.549) | (0.563) | (0.498) | (0.489) |
| n      |        | 0.052    | 0.590| 0.266  | 0.138  | 0.121  | 0.142  | 0.116  | 0.114  |
|        |        |          |      | (0.450) | (0.234) | (0.205) | (0.240) | (0.197) | (0.194) |
| M      | μB per | 1.0      | 3.134| 1.950  | 1.188  | 1.065  | 1.293  | 1.062  | 0.964  |
|        | formula |          |      | (0.622) | (0.379) | (0.340) | (0.412) | (0.340) | (0.308) |

|        |        |          |      | (0.365) | (0.320) | (0.260) | (0.347) | (0.276) | (0.248) |
Table S2. Hyper-parameter search space for CGCNN, CGCNN with sum pooling, and de-CGCNN. Parameters not mentioned here are set to the default value as in the open source codes.

| Name                      | Space          |
|---------------------------|----------------|
| atom feature length       | 32, 64         |
| hidden feature length     | 64, 128        |
| number of hidden layers   | 1, 2, 3        |
| learning rate             | 1e-3, 1e-2     |

Table S3. Hyper-parameter search space for ALIGNN, ALIGNN with sum pooling, and de-ALIGNN. Parameters not mentioned here are set to the default value as in the open source codes.

| Name                      | Space          |
|---------------------------|----------------|
| edge input feature length | 40, 80         |
| hidden feature length     | 64, 128, 256   |
| triplet input feature length | 20, 40       |
| learning rate             | 1e-3, 1e-2     |

1. Xie, T.; Grossman, J. C., Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties. *Phys Rev Lett* 2018, *120*(14), 145301.
2. Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P., Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chem. Mater.* 2019, *31*(9), 3564-3572.
3. Choudhary, K.; DeCost, B., Atomistic Line Graph Neural Network for improved materials property predictions. *npj Comput. Mater.* 2021, *7*(1).
4. Louis, S. Y.; Zhao, Y.; Nasiri, A.; Wang, X.; Song, Y.; Liu, F.; Hu, J., Graph convolutional neural networks with global attention for improved materials property prediction. *Phys Chem Chem Phys* 2020, *22*(32), 18141-18148.
5. Dunn, A.; Wang, Q.; Ganose, A.; Dopp, D.; Jain, A., Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Comput. Mater.* 2020, *6*(1), 138.
6. A. Jain*, S. P. O., G. Hautier, W. Chen, W.D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, K.A. Persson, The Materials Project: A materials genome approach to accelerating materials innovation. *APL Mater.* 2013, *1*(1), 011002.
7. Kirklin, S.; Saal, J. E.; Meredig, B.; Thompson, A.; Doak, J. W.; Aykol, M.; Rühl, S.; Wolverton, C., The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT
formation energies. *npj Comput. Mater.* **2015**, *1*(1).
8. Curtarolo, S.; Setyawan, W.; Wang, S.; Xue, J.; Yang, K.; Taylor, R. H.; Nelson, L. J.; Hart, G. L. W.; Sanvito, S.; Buongiorno-Nardelli, M.; Mingo, N.; Levy, O., AFLOWLIB.ORG: A distributed materials properties repository from high-throughput ab initio calculations. *Comput. Mater. Sci.* **2012**, *58*, 227-235.
9. Choudhary, K.; Garrity, K. F.; Reid, A. C.; DeCost, B.; Biacchi, A. J.; Walker, A. R. H.; Trautt, Z.; Hattrick-Simpers, J.; Kusne, A. G.; Centrone, A. J. a. p. a., JARVIS: An Integrated Infrastructure for Data-driven Materials Design. *arXiv:2007.01831* 2020.
10. Grosse-Kunstleve, R. W.; Sauter, N. K.; Adams, P. D., Numerically stable algorithms for the computation of reduced unit cells. *Acta Crystallogr. A* **2004**, *60*(Pt 1), 1-6.
11. WOLFF, P. M. D.; GRUBER, B., Niggli Lattice Characters. *Acta Cryst.* **1991**, *A47*, 29-36.
12. Legrain, F.; Carrete, J.; van Roekeghem, A.; Curtarolo, S.; Mingo, N., How Chemical Composition Alone Can Predict Vibrational Free Energies and Entropies of Solids. *Chem. Mater.* **2017**, *29*(15), 6220-6227.
13. Kong, S.; Ricci, F.; Guevarra, D.; Neaton, J. B.; Gomes, C. P.; Gregoire, J. M., Density of states prediction for materials discovery via contrastive learning from probabilistic embeddings. *Nat Commun* **2022**, *13*(1), 949.
14. Chen, Z.; Andrejevic, N.; Smidt, T.; Ding, Z.; Xu, Q.; Chi, Y. T.; Nguyen, Q. T.; Alatas, A.; Kong, J.; Li, M., Direct Prediction of Phonon Density of States With Euclidean Neural Networks. *Adv. Sci.* **2021**, *8*(12), e2004214.
15. Ortiz-Jiménez, G.; Modas, A.; Moosavi-Dezfooli, S.-M.; Frossard, P., Hold me tight! Influence of discriminative features on deep network boundaries. *34th Conference on Neural Information Processing Systems* 2020.
16. Cai, J.; Luo, J.; Wang, S.; Yang, S., Feature selection in machine learning: A new perspective. *Neurocomputing* **2018**, *300*, 70-79.