# Supplemental Material: Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties

Tian Xie and Jeffrey C. Grossman

*Department of Materials Science and Engineering,*
*Massachusetts Institute of Technology,*
*Cambridge, Massachusetts 02139, United States*

# I. SUPPLEMENTARY METHODS AND DISCUSSIONS

## A. Construction of crystal graphs

The connectivity between atoms in a crystal graph is determined by a method inspired by Ref. [1]. For each atom, neighbors are first searched within 6 Å radius, and they are considered as connected when they share a Voronoi face [2] with the center atom and have interatomic distance lower than the sum of the Cordero covalent bond lengths [3] with a 0.25 Å tolerance. Therefore, only strong bonding interactions are considered in the crystal graph construction.

It is worth noting that by using convolution function Eq. 5 in the main text, the connectivity between atoms becomes less important because the $\sigma(\cdot)$ part automatically ignores weak bonds. In practice, we discover that connecting 12 nearest neighbors in the initial graph construction performs as good as using the method described above.

Atom and bond properties are encoded in node feature vectors $\boldsymbol{v}_i$ and edge feature vectors $\boldsymbol{u}_{(i,j)_k}$ using one hot encoding. For discrete values, the vectors are encoded according to the category that the value belongs to; for continuous values, the range of property values is evenly divided to 10 categories and the vectors are encoded accordingly. The full list of atom and bond properties as well as their ranges are in Table S2 and Table S3. For instance, if we use the group number and period number as atom features, the atom feature vector for H will be a 27-dimensional vector with 1st and 19th element being 1 and other elements being 0. If the interatomic distance is 0.7, then the bond feature vector will be a 10-dimensional vector with 1st element being 1 and other elements being 0.

## B. Illustrative example for differentiating NaCl and KCl

We provide a simple illustrative example to explain how CGCNN works by completing the task of differentiating the structures of NaCl and KCl. Concretely, the task is to predict +1 if the crystal structure is NaCl and -1 if the structure is KCl. We can accomplish it with a CGCNN with only one convolutional layer and one pooling layer.

In Fig. S1, we convert the original crystal structures of NaCl and KCl to their corresponding crystal graphs. Note that the four Na and Cl nodes in the crystal graph of NaCl all have the same chemical environments, and we can actually simplify it to a crystal graph with two nodes. Here we use this eight-node crystal graph to illustrate that CGCNN is invariant to the choice of unit cells. For simplicity, we only use atom feature vectors without bond feature vectors in crystal graphs, since it is enough to differentiate NaCl and KCl. Specifically, each node is represented by following vectors according to the element information.

$$\boldsymbol{v}_{\mathrm{Cl}} = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}; \boldsymbol{v}_{\mathrm{Na}} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}; \boldsymbol{v}_{\mathrm{K}} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}; \tag{S1}$$

Then, we apply one convolutions to each node in the two crystal graphs according to Eq. 4 in the main text. We initialize $\boldsymbol{W}_c$ and $\boldsymbol{W}_s$ as,

$$\boldsymbol{W}_c = \begin{pmatrix} w_{c1} \\ w_{c2} \\ w_{c3} \end{pmatrix}; \boldsymbol{W}_s = \begin{pmatrix} w_{s1} \\ w_{s2} \\ w_{s3} \end{pmatrix}; \tag{S2}$$

and for simplicity, we set $\boldsymbol{b} = \boldsymbol{0}$ and $g(x) = x$. After one convolution, in NaCl, the feature of each node becomes,

$$\boldsymbol{v}_{\mathrm{Na}}^{(1)} = 6w_{c1} + w_{s2} \tag{S3}$$

$$\boldsymbol{v}_{\mathrm{Cl}}^{(1)} = 6w_{c2} + w_{s1} \tag{S4}$$

and in KCl,

$$\boldsymbol{v}_{\mathrm{K}}^{(1)} = 8w_{c1} + w_{s3} \tag{S5}$$

$$\boldsymbol{v}_{\mathrm{Cl}}^{(1)} = 8w_{c3} + w_{s1} \tag{S6}$$

After convolution, we apply a simple normalized pooling by summing over the feature vectors of all nodes $\boldsymbol{v}_i^{(1)}$ within the crystal graph, and then divide it by the total number of nodes. Consequently, the overall feature vectors of the two crystal graphs are,

$$\boldsymbol{v}_{\mathrm{NaCl}} = 3w_{c1} + 3w_{c2} + 0.5w_{s1} + 0.5w_{s2} \tag{S7}$$

$$\boldsymbol{v}_{\mathrm{KCl}} = 4w_{c1} + 4w_{c3} + 0.5w_{s1} + 0.5w_{s3} \tag{S8}$$

Since the crystal feature vectors are already one-dimension, we do not need another output layer to map them into the target value. So, our predictions are,

$$\hat{y}_{\mathrm{NaCl}} = 3w_{c1} + 3w_{c2} + 0.5w_{s1} + 0.5w_{s2} \tag{S9}$$

$$\hat{y}_{\mathrm{KCl}} = 4w_{c1} + 4w_{c3} + 0.5w_{s1} + 0.5w_{s3} \tag{S10}$$

We can easily find $\boldsymbol{W}_c$ and $\boldsymbol{W}_s$ that make $\hat{y}_{\mathrm{NaCl}} = 1$ and $\hat{y}_{\mathrm{KCl}} = -1$, showing that CGCNN are capable of differentiating NaCl and KCl. In this example, the existence of certain weights $w_{ci}$ or $w_{si}$ in Eq. S9 and Eq. S10 indicates the occurrence of elements as centers or neighbors respectively, while the factor represents the frequency of occurrence, both of which can help differentiating the two crystals.

When more training data is available, we could not find $\boldsymbol{W}_c$ and $\boldsymbol{W}_s$ that result in zero loss since the weights are shared for all crystals. Methods like stochastic gradient descent (SGD) can me used to minimize the loss and find the weights that maximize the prediction performance. Also, more complex CGCNN structures can be used to capture more structure information.

### C. Hyperparameter optimization

The hyperparameters are parameters that defines the CGCNN. They include graph parameters that are used to generate the crystal graph, architecture parameters that are used to define the convolutional neural network on top of the crystal graph, and training parameters that are used for the training process. Unlike the weights that are trained via SGD, the hyperparameters are chosen through a train-validation process.

We first randomly divide our database into three parts: training set (60%), validation set (20%), and test set (20%). Model with different hyperparameters are trained on training set via SGD, and the resulting weights are used to predict the property of crystals in validation set. By comparing to the properties calculated by DFT, and the hyperparameters that provide lowest MAE or AOC in the validation set are chosen as the optimum hyperparameters.

For convolution function Eq. 4, we use random search to optimize all the hyperparameters listed in Table S1. Since the influence of hyperparameter optimization is much smaller than the convolution functions as shown in Fig. S3, we only optimize number of convolutional layers, regularization term, and step size of the Adam optimizer when using convolution function Eq. 5.

### D. Pooling layer choices

Throughout this work, normalized summation is used as a pooling function as described in the main text. However, for convolutional function Eq. 4, we sum up different feature vectors for the purpose of either maximizing prediction performance or interpretability.

To maximize prediction performance, we utilize feature vectors from all convolutional layers. Each feature vector $\boldsymbol{v}_i^{(t)}$ from the $t$-th convolution is first transformed to the same dimension by a linear map and then sparsified by a Softmax function (Eq. S12), resulting in a vector $\tilde{\boldsymbol{v}}_i^{(t)}$ with the same dimension. Then, the pooling layer sums over feature vectors $\tilde{\boldsymbol{v}}_i^{(t)}$ from all convolutional layers where $t = 0, 1, .., R$ and over all atoms $i = 1, 2, ..., N$. The resulting crystal vector $\boldsymbol{v_c}$ is then normalized to be invariant to crystal size. This pooling includes all intermediate feature vectors representing chemical environments of various radius, which is more complete and resulting better prediction performance.

$$\mathrm{Softmax}\left(\boldsymbol{z}\right)_j = \frac{\exp(z_j)}{\sum_k \exp(z_k)} \tag{S11}$$

$$\tilde{\boldsymbol{v}}_i^{(t)} = \mathrm{Softmax}(\boldsymbol{W}_t \boldsymbol{v}_i^{(t)} + \boldsymbol{b}_t) \tag{S12}$$

$$\boldsymbol{v_c} = \sum_{t,i} \tilde{\boldsymbol{v}}_i^{(t)} \tag{S13}$$

To get better interpretability, we directly map the final feature vector after $R$ convolutional layers and $L_1$ hidden layers $\boldsymbol{v}_i^{(R)}$ to a scalar $\tilde{v}_i$ using a linear transform (Eq. S14), and then summing up the scalars to predict the target property $v_c$. After training with the global property $v_c$, the model automatically learns the contribution of each local chemical environments represented by $\tilde{v}_i$.

$$\tilde{v}_i = \boldsymbol{W}^T \boldsymbol{v}_i^{(R)} + b \tag{S14}$$

$$v_c = \frac{1}{N} \sum_i \tilde{v}_i \tag{S15}$$

For convolution function Eq. 5, we directly compute the normalized sum of the last feature vector $\boldsymbol{v}_i^{(R)}$ after R convolutional layers as a pooling function (Eq. S16) since the residual structure in Eq. 5 allows information passed to the last layer which makes adding $\boldsymbol{v}_i^{(t)}$ from previous convolutional layers unnecessary.

$$\boldsymbol{v}_c = \frac{1}{N}\sum_i \boldsymbol{v}_i^{(R)} \tag{S16}$$

## II. SUPPLEMENTARY FIGURES

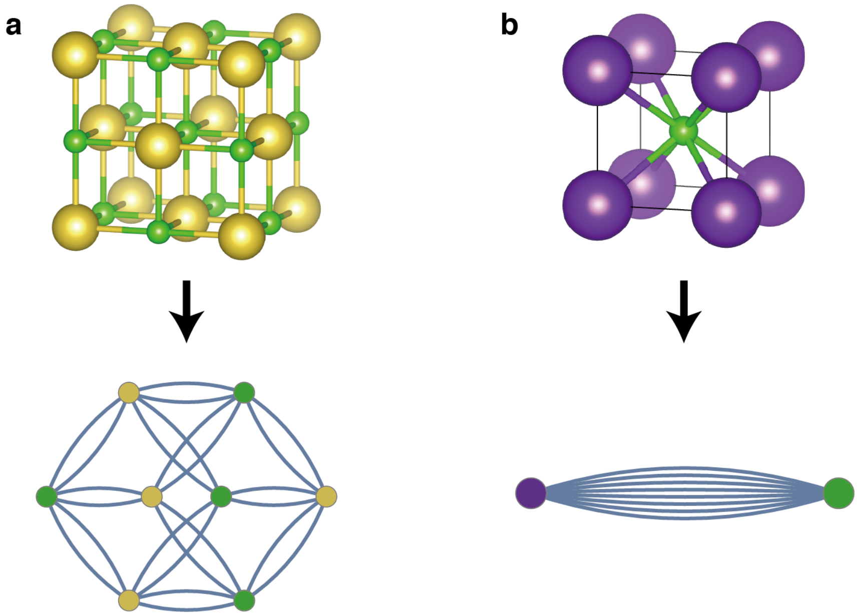

FIG. S1: The crystal structures and crystal graphs of NaCl (a) and KCl (b).

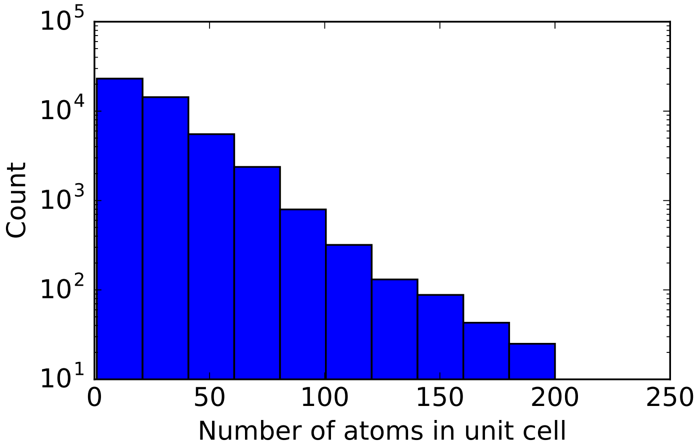

FIG. S2: Histogram representing the distribution of the number of atoms in the primitive cells.

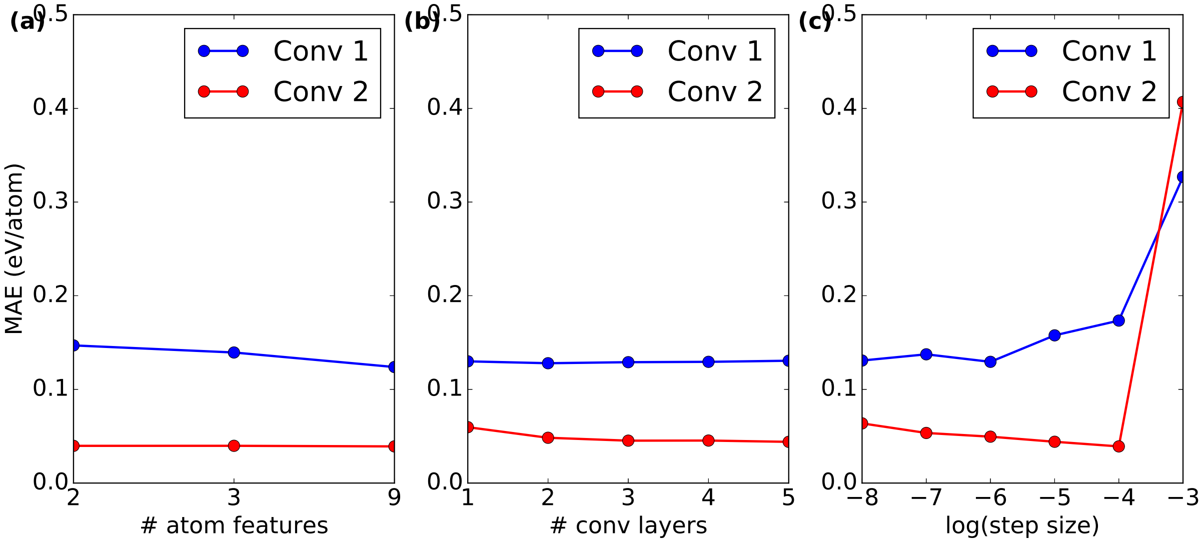

FIG. S3: The effect of different hyperparameters on the validation mean absolute errors (MAEs). The blue points denotes models using convolution function Eq. 4, and the red points denotes models using Eq. 5. (a) Number of atom features. The 2 features include group number and period number, the 3 features additionally include electronegativity, and the 9 features include all properties in Table S2. (b) Number of convolutional layers. (c) Logarithm of the step size.

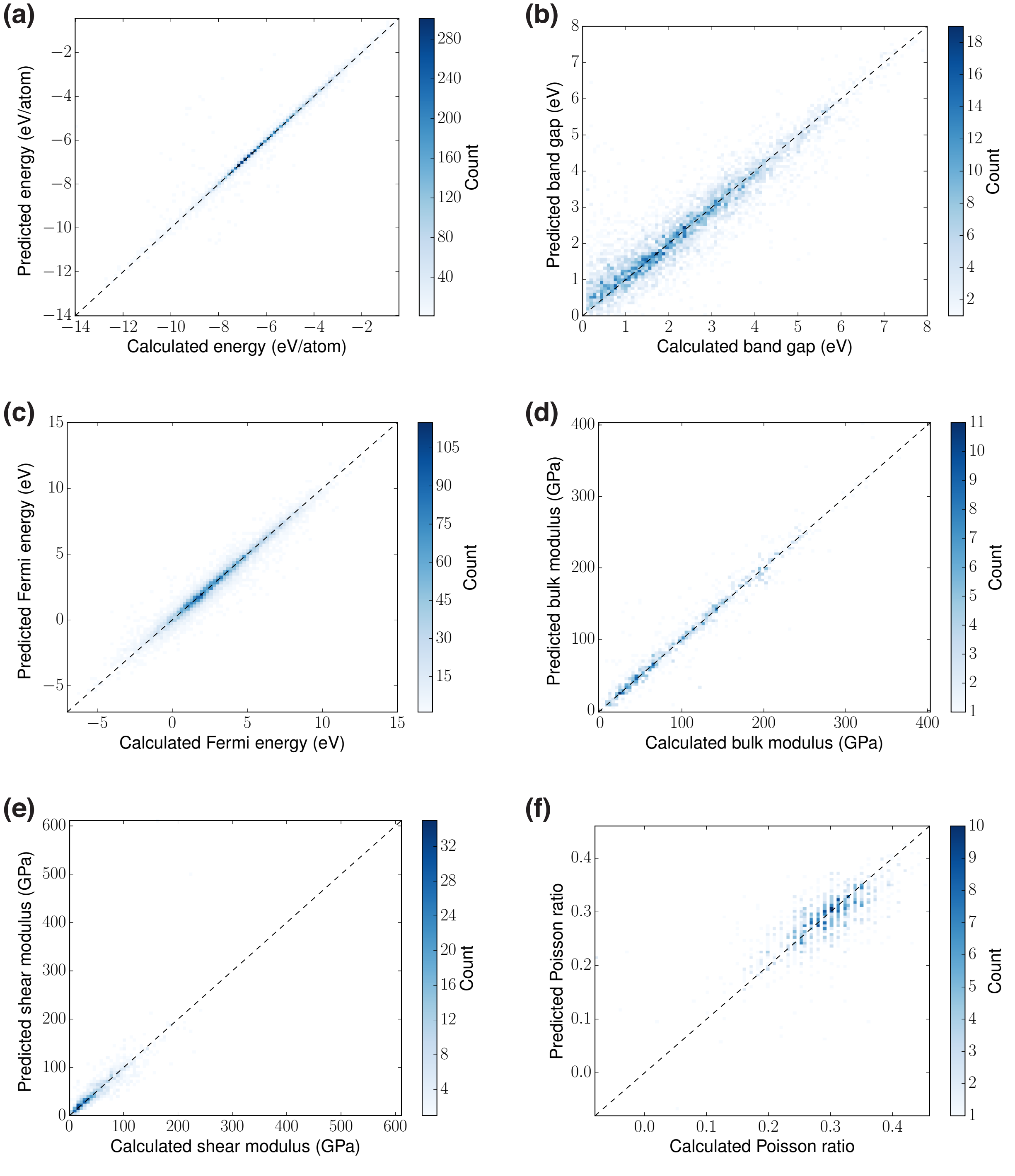

FIG. S4: 2D histogram visualizing the predictive performance of six properties. (a) Total energy per atom. (b) Band gap. (c) Fermi energy. (d) Bulk moduli. (e) Shear moduli. (e) Poisson ratio.

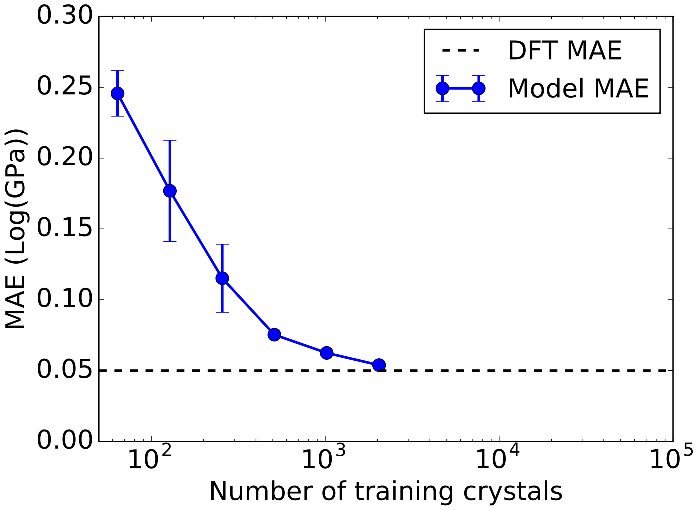

FIG. S5: The MAE of predicted bulk modulus with respect to DFT values against the number of training crystals. The dashed line shows the MAE of DFT calculations with respect to experimental results [4], which is 0.050 Log(GPa).

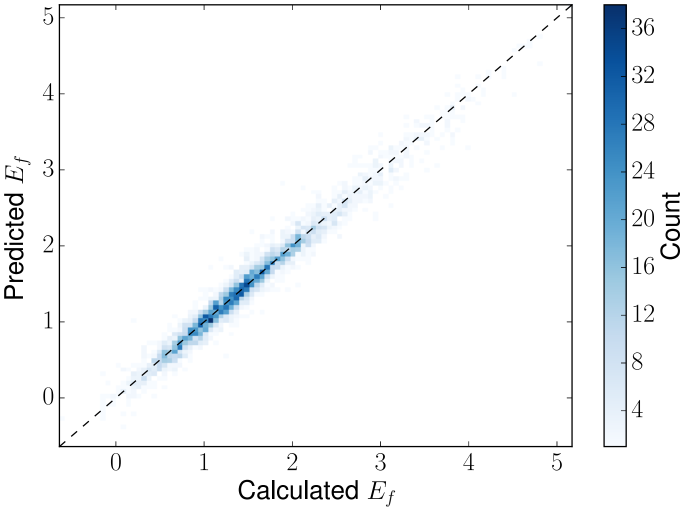

FIG. S6: 2D histogram visualizing the performance of predicting formation energy of perovskites using a full pooling layer with Eq. 4 as the convolution function.

## III. SUPPLEMENTARY TABLES

TABLE S1: A list of hyperparameters that are optimized in this work.

| Hyperparameter | Range |
|---|---|
| Number of properties used in atom feature vector $\boldsymbol{v}_i$ | 2, 3, 10 |
| Number of convolutional layers | 1, 2, ..., 5 |
| Length of learned atom feature vector $\boldsymbol{v}_i^{(t)}$ | 10, 20, 50, 100, 200 |
| Number of hidden layer $L_1$[a] | 1, 2, ..., 4 |
| Number of hidden layer $L_2$[a] | 1, 2, ..., 5 |
| Regularization term $\lambda$[b] | $\mathrm{e}^{-6}$, $\mathrm{e}^{-4}$, $\mathrm{e}^{-2}$, $\mathrm{e}^{0}$ |
| Scaling factor of the Gaussian initialization of weights | $\mathrm{e}^{-8}$, $\mathrm{e}^{-6}$, $\mathrm{e}^{-4}$, $\mathrm{e}^{-2}$ |
| Step size of the Adam optimizer[5] | $\mathrm{e}^{-8}$, $\mathrm{e}^{-7}$, $\mathrm{e}^{-6}$, $\mathrm{e}^{-5}$, $\mathrm{e}^{-4}$, $\mathrm{e}^{-3}$ |
| Dropout fraction[6] | 0, 0.1, 0.2 |

[a] The hidden layers $L_1$ and $L_2$ are not used simultaneously in this work. We only use $L_2$ for the prediction of material properties and $L_1$ for the learning of individual site energies in perovskites.

[b] $L^2$ regularization term $\lambda \|\boldsymbol{W}\|_2^2$ is added to cost function to reduce overfitting.

TABLE S2: Properties used in atom feature vector $\boldsymbol{v}_i$

| Property | Unit | Range | # of categories |
|---|---|---|---|
| Group number | – | 1,2, ..., 18 | 18 |
| Period number | – | 1,2, ..., 9[a] | 9 |
| Electronegativity[7, 8] | – | 0.5–4.0 | 10 |
| Covalent radius[3] | pm | 25–250 | 10 |
| Valence electrons | – | 1, 2, ..., 12 | 12 |
| First ionization energy[9][b] | eV | 1.3–3.3 | 10 |
| Electron affinity[10] | eV | -3–3.7 | 10 |
| Block | – | s, p, d, f | 4 |
| Atomic volume[b] | $cm^3$/mol | 1.5–4.3 | 10 |

[a] The lanthanide and actinide elements are considered as period 8 and 9 respectively.

[b] Log scale is used for these properties.

TABLE S3: Properties used in bond feature vector $\boldsymbol{u}_{(i,j)_k}$

| Property | Unit | Range | # of categories |
|---|---|---|---|
| Atom distance | Å | 0.7–5.2 | 10 |

TABLE S4: Comparison of the prediction performance of seven different properties on test sets using different convolution functions.

| Property | # Train data | Unit | $\mathrm{MAE}_{\mathrm{Eq.\ 4}}$ | $\mathrm{MAE}_{\mathrm{Eq.\ 5}}$ |
|---|---|---|---|---|
| Formation energy | 28046 | eV/atom | 0.112 | 0.039 |
| Absolute energy | 28046 | eV/atom | 0.227 | 0.072 |
| Band gap | 16458 | eV | 0.530 | 0.388 |
| Fermi energy | 28046 | eV | 0.564 | 0.363 |
| Bulk moduli | 2041 | Log(GPa) | 0.084 | 0.054 |
| Shear moduli | 2041 | Log(GPa) | 0.113 | 0.087 |
| Poisson ratio | 2041 | – | 0.033 | 0.030 |

TABLE S5: Perovskites with energy above hull lower than 0.2 eV/atom discovered using combinational search.

| Formula | A site | B site | Formation energy per atom (eV/atom) |
|---|---|---|---|
| Training Set (60%) | | | |
| TlNbO3 | Tl | Nb | 0.0 |
| SnTiO3 | Sn | Ti | 0.1 |
| PbVO3 | Pb | V | 0.04 |
| SnTaO3 | Sn | Ta | 0.0 |
| TlWO3 | Tl | W | 0.12 |
| PbMoO3 | Pb | Mo | 0.18 |
| PbCrO3 | Pb | Cr | 0.14 |
| SnNbO3 | Sn | Nb | 0.14 |
| SnTaO2N | Sn | Ta | 0.14 |
| TlTaOFN | Tl | Ta | 0.18 |
| TlTaO2F | Tl | Ta | 0.04 |
| TlHfO2F | Tl | Hf | 0.18 |
| PbTiO3 | Pb | Ti | 0.06 |
| InNbO3 | In | Nb | 0.06 |
| InWO3 | In | W | 0.18 |
| InTaO3 | In | Ta | -0.16 |
| InNbO2F | In | Nb | 0.18 |
| InTaO2S | In | Ta | 0.18 |
| Validation Set (20%) | | | |
| TlNbO2F | Tl | Nb | 0.08 |
| TlZrO2F | Tl | Zr | 0.14 |
| SnVO3 | Sn | V | 0.12 |
| TlTiO2F | Tl | Ti | -0.02 |
| PbNbO2N | Pb | Nb | 0.18 |
| PbZrO3 | Pb | Zr | 0.08 |
| PbNbO3 | Pb | Nb | 0.04 |
| Test Set (20%) | | | |
| BiCrO3 | Bi | Cr | 0.14 |
| PbVO2F | Pb | V | 0.14 |
| SnNbO2N | Sn | Nb | 0.18 |
| PbHfO3 | Pb | Hf | 0.1 |
| TlTaO3 | Tl | Ta | 0.1 |
| PbTaO3 | Pb | Ta | 0.18 |
| InTaO2F | In | Ta | 0.08 |
| InZrO2F | In | Zr | 0.16 |

[1] O. Isayev, C. Oses, C. Toher, E. Gossett, S. Curtarolo, and A. Tropsha, Nature Communications **8** (2017).

[2] V. A. Blatov*, Crystallography reviews **10**, 249 (2004).

[3] B. Cordero, V. Gómez, A. E. Platero-Prats, M. Revés, J. Echeverría, E. Cremades, F. Barragán, and S. Alvarez, Dalton Transactions , 2832 (2008).

[4] M. De Jong, W. Chen, T. Angsten, A. Jain, R. Notestine, A. Gamst, M. Sluiter, C. K. Ande, S. Van Der Zwaag, J. J. Plata, *et al.*, Scientific data **2**, 150009 (2015).

[5] D. Kingma and J. Ba, arXiv preprint arXiv:1412.6980 (2014).

[6] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, Journal of Machine Learning Research **15**, 1929 (2014).

[7] R. Sanderson, Science **114**, 670 (1951).

[8] R. Sanderson, Journal of the American Chemical Society **74**, 4792 (1952).

[9] A. Kramida, Y. Ralchenko, J. Reader, *et al.*, National Institute of Standards and Technology, Gaithersburg, MD (2013).

[10] W. M. Haynes, *CRC handbook of chemistry and physics* (CRC press, 2014).