Supplementary Materials for Crystal Hypergraph Convolutional Neural Networks

Alexander J. Heilman, Weiyi Gong, Qimin Yan

S1 Motif Features: Structure Order Parameters & Continuous Symmetry Measures

The geometry of the motifs were incorporated as features composed of a concatenated list of structure order parameters and continuous symmetry measures (CSMs) for a set of common local environments.

Structure order parameters are coordinate system invariant measures of 3 dimensional structure that are designed to be close to one when a given structure is similar to some prototypical arrangement. Note that this isn't in general a true 'distance'-like measure to some shape as a CSM is, however. A CSM is essentially defined so that it may act as a 'distance' from some prototypical shape to some given structure. The list of order parameters included as motif features are those that implemented in existing pymatgen code and described in [1, 2].

S2 CHGConv

A specific implementation of a hypergraph convolutional operator in the hypergraph message passing framework is a generalization of CGConv implemented in pytorch geometric and based on CGCNN's convolutional operator defined in eq (5) of the original paper.

$$\begin{aligned} x_i^{t+1} &= \sum_{b_j} f(x_i^t, b_j, \mathrm{AGG}(\{x_j^t \in b_j\})) \\ &= \mathrm{BN}\Big[\sum_{b_j} \sigma\big(W_c \cdot [x_j \oplus b_j \oplus \mathrm{AGG}(\{x_j^t \in b_j\}])\big) \\ &\qquad \cdot S^+(W_f \cdot (x_j \oplus b_j \oplus \mathrm{AGG}(\{x_j^t \in b_j\})))\Big] \end{aligned}$$

For the model utilized in this work, the AGG function chosen was a combined set of component-wise maximum, minimum, average, and standard deviation, all with learnable attention weights. This combined use of multiple aggregation functions was inspired by a similar approach taken in the *ChemGNN* model [3].

S3 Hyperparameters for Testing

For each convolutional structure, testing was done for a model with 3 convolutional layers. Each convolutional layer consists of back-to-back convolution from the smallest to the largest hyperedge type (for example two bond & motif layers consist of a total sequence of bond, motif, bond and motif convolution).

Hyperparameter	**Value**
Node Hidden Feature Dimension	64
Post-Convolution Linear Width	128
Number of Convolutional Layers	3
Number of Epochs	300
Batch-size	64 ($< 20,000$ Samples) *or* 128
Optimizer	SGD
Learning Rate (Epoch <150)	0.01
Learning Rate (Epoch >150)	0.001

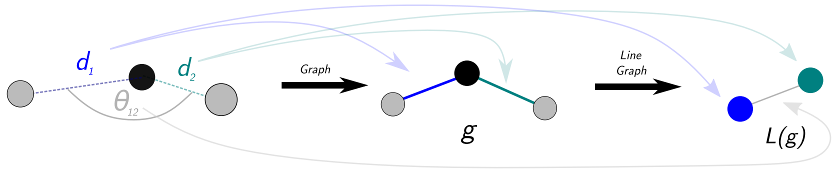
Stochastic gradient descent (SGD) was used as an optimizer through training with an initial learning rate of 0.01. A multi-step learning rate scheduler divided this learning rate by a factor of 10 at epoch 150, with training running for a total of 300 epochs.

Hidden node features were of dimension 64 through all convolutional layers, and a hidden output layer of dimension 128 was used (similar to CGCNN's architecture). The loss functions utilized were Mean Squared Error (MSE, for regression tasks) and cross entropy (for classification tasks). Accuracy is then reported in Mean Absolute Error (MAE) for regression tasks and area under curve (AUC) for classification tasks.

Results reported were averaged over 5 folds of nested cross-validation. The datasets were divided into 80% for training and 20% for test for each fold, with a further 20% of the training subset being used as an indicative validation set, where the best performance on this dataset was used to select the model applied to the test set.

S4 Comparison to Line Graph

A more usual approach for the incorporation of bond angle information is via the construction of a line graph, as in [4, 5].



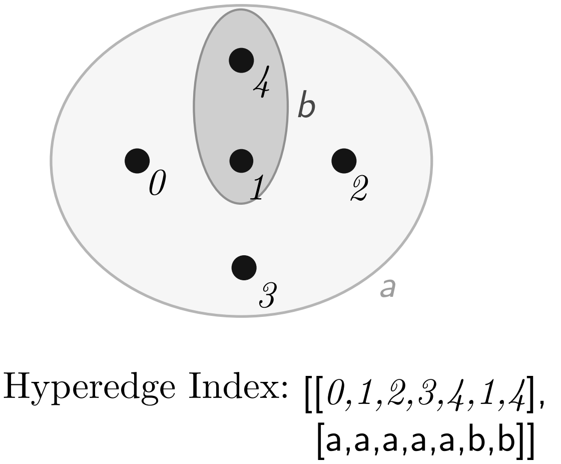
These models generally first update the edge features of the crystal graph $\mathcal{G}$ by first applying some graph convolutional operator to the line graph $L(\mathcal{G})$ with angles encoded in $L(\mathcal{G})$'s initial edge features.

Our argument against such representation schemes here is that the order of messages grows combinatorically for derived line graphs as $\mathcal{O}(nm^2)$, where n is the number of nodes and m is the average number of edges per node in $\mathcal{G}$.

Here, we incorporate a similar level of higher-order geometrical structure instead in a local environment, or 'motif', hyperedge (defined below). Note that these include only an extra number of messages on the order $\mathcal{O}(mn)$ if each node in a motif gets a message, or on the order $\mathcal{O}(n)$ if only center nodes are updated by their own motif hyperedges.

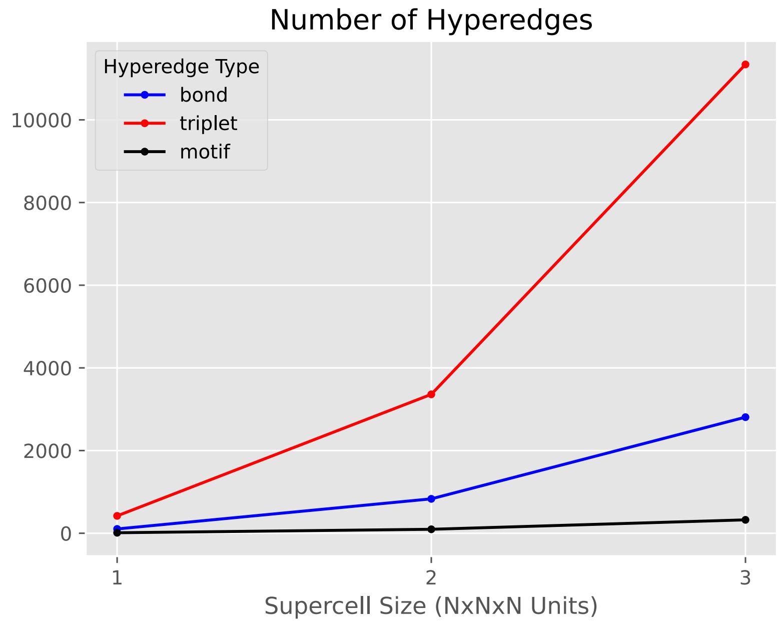
S5 Hyperedge Index

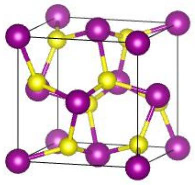
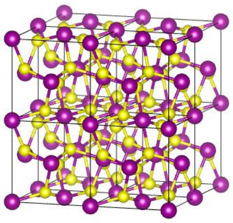
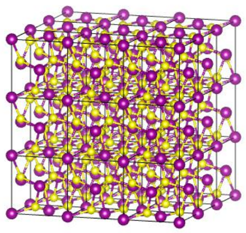
Hypergraphs are treated as a set of node feactures x, hyperedge features h, and hyperedge indices I. The hyperedge index is, computationally, treated as a $[2, nm]$ dimensional vector (where m is the number of hyperedges and n is the avereage number of nodes contained in any hyperedge). The first index is the node contained and the second index is the containing hyperedge (as in [6]).



S6 Message Number Scaling for Different Hyperedge Types

The rationale for including motifs in lieu of triplets (as in line graphs) is most poignant when considering material systems of increasing unit-cell size. To make this point most clear, here we consider the number of hyperedges required for the three different types of hyperedges considered in this work for different sized supercells of Manganese Oxide (MnO_2).



Conventional Cell	$2 \times 2 \times 2$ Super Cell	$3 \times 3 \times 3$ Super Cell
		
Num. of Bonds 104 Num. of Triplets 420 Num. of Motifs 12	Num. of Bonds 832 Num. of Triplets 3360 Num. of Motifs 96	Num. of Bonds 2808 Num. of Triplets 11340 Num. of Motifs 324

Here, the number of triplets clearly grows exponentially, whereas the number of bonds grows quadratically, and the number of motifs, linearly. As such, additional geometric resolution may be afforded by relatively few motif hyperedges, as compared to triplet-based constructions.

S7 Case Study on Discrimination of Similar Environments

To demonstrate the importance of motif-level hyperedges in applications to material systems, the node embeddings of two compositionally-similar, but structurally distinct materials are considered here: T-phase and H-phase MoS_2. To demonstrate the effectiveness of the motif models in sooner recognizing structural differences, the difference in these node embeddings are compared within these material sets after 1 through 3 convolutional layers of either bond-, motif-, or triplet-level hyperedges. Node representation difference was calculated by subtracting the normalized dot-product of the central atom's node embeddings from one.

While bond-only models could generally distinguish node embeddings of central atoms between different, but similar, structures after a few layers, motifs most often capture the structural differences earliest (that is, the one-layer models of motifs always show the greatest difference). However, it should be noted this metric is unreliable, insofar that neural networks are essentially black boxes and hence sheer differences in node embeddings are hard to translate directly to model performance. The ambiguity of this metric is made clear in the difference for 1H- vs 1T-MoS_2, where the initial motif features are orthogonal, but the node representation difference is minimal after the first layer.

S8 Comparison of Order Parameters and Continuous Symmetry Measures as Motif Features

To compare the importance of motif features, the data set that most benefited from motif features were tested both with and without the two primary types of motif features considered in this work: Local Structure Order Parameters (LSOPs) as defined in [1, 2]; and Continuous Symmetry Measures [7, 8] for 59 common coordination polyhedra. Bond and motif convolutional models were tested on this bulk moduli dataset for 500 epochs with no motif features, only CSMs, only LSOPs, and both feature sets concatenated. For all tests, the batch size was 128, Stochastic Gradient Descent was used as the optimizer with an initial learning rate of 0.1, with a reduced learning rate of 0.01 being applied at epoch 300. These results are tabulated in S8.

Motif	**Validation MAE**
Features	$\mathrm{Log}_{10}(K_{vrh})$
None	0.777
CSMs	0.758
LSOPs	0.785
Both	0.772

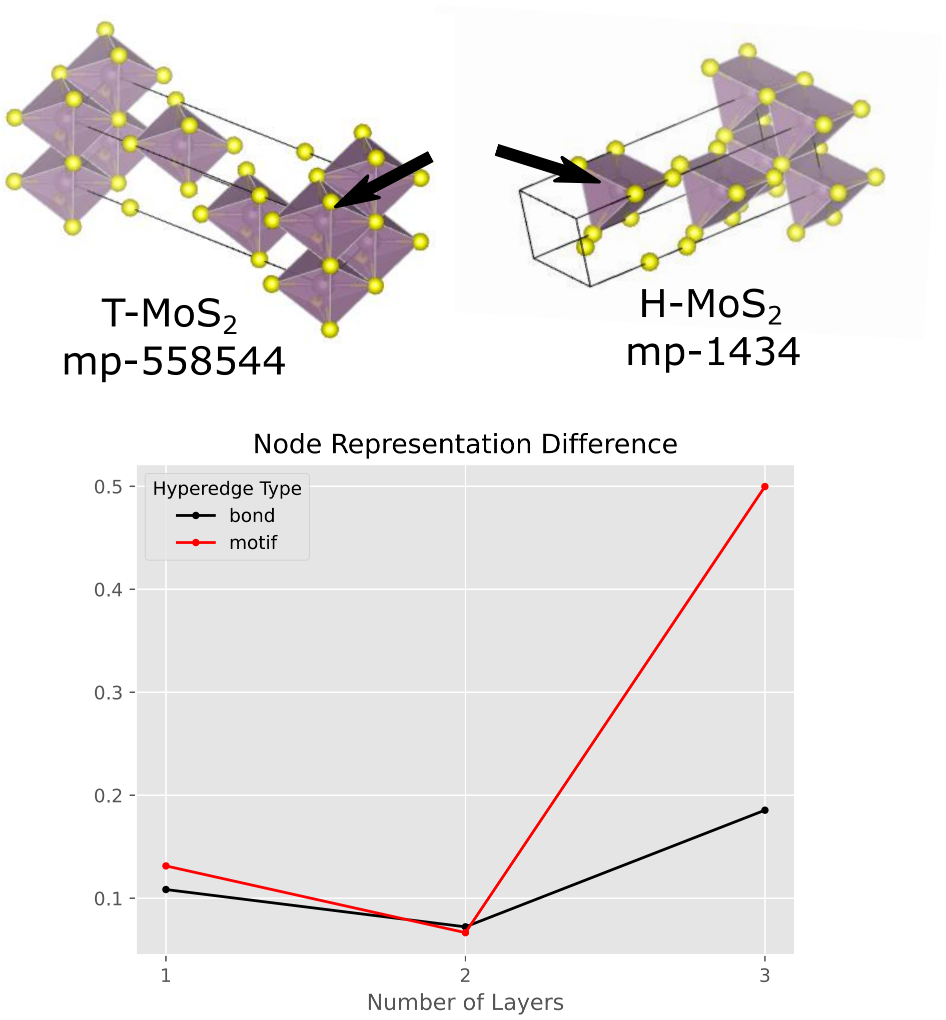


Figure S1: Difference in node representations after 1-3 layers of either bond-, motif-, or triplet-only layers as determined by one minus the normalized dot product of central node representations for two phases of MoS_2, a common Transition Metal Dihalcogenide (TMD).

Suprisingly, the LSOP features negatively impact performance, both as the sole motif features and as compliments to CSM features. CSM-only features performed best on the validation set, suggesting they have offer better generalizatbility as a feature. This may be due to the fact that CSM-based features result in sparse (similar to one-hot) encodings, whereas LSOP features tend more towards scalar-valued features. That is, while both CSMs and LSOPs are scalar-valued, a large subset of incommensurate ideal shapes used in CSM calculations results in many zero entries, whereas LSOPs may generally be computed regardless and just return smaller scalar values, so that the sparse CSM features may more adequately distinguish environments through the model.

S9 Performance on Matbench Folds

Results tabulated in the main text are averaged over 5 folds of nested cross-validation, with test indexes supplied by MatBench [9]. Accordingly, the uncertainty in test performance Δx is calculated as:

$$\Delta x = \frac{R}{2\sqrt{N}} \tag{1}$$

with R the range of values and N the number of runs (here, 5 for the 5 folds). Furthermore, the test set performance for each fold is tabulated below.

MP - Band Gap			
Fold	Bond-only	Bond & Motif	Bond & Triplet
1	0.353	0.367	0.338
2	0.360	0.374	0.323
3	0.360	0.370	0.317
4	0.357	0.356	0.321
5	0.351	0.363	0.323

MP - Formation Energy			
Fold	Bond-only	Bond & Motif	Bond & Triplet
1	0.353	0.367	0.338
2	0.360	0.374	0.323
3	0.360	0.370	0.317
4	0.357	0.356	0.321
5	0.351	0.363	0.323

MP - Metalicity			
Fold	Bond-only	Bond & Motif	Bond & Triplet
1	0.353	0.367	0.338
2	0.360	0.374	0.323
3	0.360	0.370	0.317
4	0.357	0.356	0.321
5	0.351	0.363	0.323

$\mathrm{Log}_{10}(\mathrm{K}_{vrh})$			
Fold	Bond-only	Bond & Motif	Bond & Triplet
1	0.353	0.367	0.338
2	0.360	0.374	0.323
3	0.360	0.370	0.317
4	0.357	0.356	0.321
5	0.351	0.363	0.323

$\mathrm{Log}_{10}(\mathrm{G}_{vrh})$			
Fold	Bond-only	Bond & Motif	Bond & Triplet
1	0.353	0.367	0.338
2	0.360	0.374	0.323
3	0.360	0.370	0.317
4	0.357	0.356	0.321
5	0.351	0.363	0.323

$\mathrm{Log}_{10}(\mathrm{K}_{vrh})$			
Fold	Bond-only	Bond & Motif	Bond & Triplet
1	0.353	0.367	0.338
2	0.360	0.374	0.323
3	0.360	0.370	0.317
4	0.357	0.356	0.321
5	0.351	0.363	0.323

References

[1] N. E. R. Zimmermann, M. K. Horton, A. Jain, and M. Haranczyk, *Front. Mater.* **4**, 34 (2017).

[2] N. E. Zimmermann and A. Jain, *RSC Adv.* **10**, 6063–6081 (2020).

[3] C. Chen, E. Xu, D. Yang, C. Yan, T. Wei, H. Chen, Y. Wei, and M. Chen, *Neural Comput. Apl.* **37**, 3287–3301 (2024).

[4] K. Choudhary and B. DeCost, *npj Comput. Mater.* **7**, 1–8 (2021).

[5] C. Chen and S. P. Ong, *Nat. Comput. Sci.* **2**, 718–728 (2022).

[6] S. Bai, F. Zhang, and P. H. Torr, *Pattern Recogn.* **110**, 107637 (2021).

[7] M. Pinsky and D. Avnir, *Inorg. Chem.* **37**, 5575–5582 (1998).

[8] J Lima-de Faria, E Hellner, F Liebau, E Makovicky, and E Parthé, *Acta Crystallogr. A* **46**, 1–11 (1990).

[9] A. Dunn, Q. Wang, A. Ganose, D. Dopp, and A. Jain, *npj Comput. Mater.* **6**, 138 (2020).