Crystal Hypergraph Convolutional Networks

Alexander J. Heilman, Weiyi Gong, and Qimin Yan

Department of Physics
Northeastern University, 360 Huntington Ave, Boston, MA 02115

February 3, 2025

Abstract

Graph representations of solid state materials that encode only interatomic-distance information lack geometrical resolution, resulting in degenerate representations that may map distinct structures to equivalent graphs. Here we propose a hypergraph representation scheme for materials that allows for the association of higher-order geometrical information with hyperedges. Hyperedges generalize edges to connected sets of more than two nodes, and may be used to represent triplets and local environments of atoms in materials. This generalization of edges requires a different approach in graph convolution, three of which are developed in this work. These crystal hypergraph convolutional networks are trained based on various property prediction tasks for a vast set of solid-state materials in the Materials Project and MatBench databases. Results presented here focus on the improved performance of models based on both pair-wise edges and local environment hyperedges. These results demonstrate that hypergraphs are an effective and efficient method for incorporating geometrical information in material representations.

Introduction

Machine learning has proven to be a computationally cost-effective and powerful predictive tool in the screening of large sets of material systems for certain material properties [31, 34, 18, 19]. Some of the most effective state-of-the-art models applied to invariant target predictions represent material systems as graphs [27, 35, 6, 4, 8, 21]. These graphs encode physical properties in feature vectors associated with graph components, and update or 'learn' these features with a trained graph neural network or message passing network [15].

One problem with such graphical representations, however, is the lack of representation of higher-order geometrical structure, since the constructed crystal graphs can only include pair-wise descriptors. This may make it difficult or even impossible for models to distinguish between compositionally similar but structurally distinct systems with unique material properties [24]. Other works have approached this problem by including higher-order geometrical features such as overlapping bonds' angles [9, 5, 24]. However, these approaches come with an increase in the total number of messages exchanged through convolution that is quadratic with respect to $\bar{N}_{\text{edges}}$, the average number of edges per atom.

Here, we propose the concept of *crystal hypergraphs* to address this lack of geometrical information in the more restrictive graph representations. In a crystal hypergraph, we may define larger (than strictly pair-wise) hyperedges that correspond to higher-order geometrical structures of material systems explicitly, such as triplets of neighboring atoms, or coordination polyhedra/motifs [22, 17, 33, 11, 37, 28]. These different structures then may have different coordinate invariant features associated with them, such as angles and local order parameters [39, 40, 26], respectively. Note that in that regard, crystal hypergraphs are naturally heterogeneous in their hyperedges, since there are different feature sets for different types of hyperedges.

Of course, the definition of a more general hypergraph representation requires the generalization of the message passing framework mentioned above. Here, we propose three possible approaches to such a generalization that handle the now-variable size of hyperedges. In a certain sense, these allow for the learning of a certain type of 'cluster-correlation expansion' [3, 30, 25] by the model, where clusters of interest correspond to the hyperedges defined.

As a proof of concept, we propose and implement a crystal hypergraph convolutional model (CHGCNN) that incorporates invariant geometric features for bonds, triplets, and motifs in crystals as hyperedge features. This provides us a unique

opportunity to demonstrate the importance of different order structures for these different material property prediction tasks. Namely, we compare the performance of models based on atom, bond, and triplet information against those incorporating atom, bond, and motif information (i.e. first shell hyperedges) on various predictive tasks with varying data sizes.

Results presented here indicate that first-shell (motif) hyperedges may be sufficient, if not more informative, than triplet hyperedges for many common predictive tasks. This comes at a substantially lower computational cost, in terms of the total number of messages exchanged through graph convolution.

The structure of this work is as follows. We give a brief overview of crystal graph construction and message passing networks. A motivating representation problem is then identified with our definitions and the concept of crystal hypergraphs is introduced, with a particular focus on different types of hyperedges and their corresponding feature sets. Three generalized message passing frameworks are then considered, and a specific model architecture is presented. Finally, this specific architecture is used on various datasets to evaluate the performance of different sets of hyperedge types.

Background

Crystal Graphs

A common representation of crystalline systems in machine learning is via graphs (that is, collections of nodes and binary connections between them). We may define a crystal graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ as a set of vertices $v_i \in \mathcal{V}$, corresponding to each atom i, and edges $e_{ij} \in \mathcal{E}$, where edges are determined by some physical criteria. Physical information is then associated with the objects in these graphs by way of feature vectors. These are vectors with components describing the physical characteristics of their corresponding graph component, and which may be further 'learned' or updated through a graph neural network.

A commonly applied criterion for the formation of edges between atoms is a combination of a maximum distance cutoff r_{max} and a maximum number of neighbors for each node N_{max}. That is, for each atom, edges are constructed between its node and its $\leq N_{max}$-th closest neighbors in the crystalline structure within a shell of radius r_{max}.

The nodes' feature vectors encode the atomic information of the sites they describe. Two usual techniques include: explicitly engineered feature vectors (as in [35]); and the learning of encodings for atomic sites based only on their atomic number (as in [6]), beginning with some random initialization. Edge features are often derived exclusively from their distance.

Crystal graphs are usually constructed solely for use in graph convolutional neural networks. Perhaps the most general framework in which we may define graph convolution is the message passing framework, defined by Gilmore [15]. A message passing network updates nodes based on 'messages' generated by the features of, and passed through, neighboring nodes (that is, nodes sharing an edge).

However, the construction above lacks higher-order geometrical information within the crystals. That is, distinct local geometrical environments of atoms (motifs) may be mapped degenerately to the same crystal graph. As a simple example of the low resolution manifest in crystal graphs, consider two atomic systems below: one with a local cubic symmetry, and another with a square anti-prism local environment; but both with the same bonding atoms. As demonstrated in Figure 1 both structures would map to the same crystal graph, but could be easily distinguished with an additional descriptor describing the local geometry of each central atom.

Alternatively, one could include atomic position in the node features or a vector direction in edge features. However, this generally requires a unique treatment of such coordinate-system dependent information through convolution if the output is to maintain *invariance* with respect to changes in the coordinate system. As such, often only coordinate system invariant features are included in crystal graph representations, such as distance and atomic properties.

Methods

Crystal Hypergraph Construction

The method proposed here reduces the intrinsic limitations of invariant crystal graph featurs by allowing the explicit incorporation of higher-order geometrical information in the form of hyperedges, which can be used to directly represent these higher-order structures.

A crystal hypergraph $\mathcal{H} = \{\mathcal{V}, H\}$ is a collection of nodes $v_i \in \mathcal{V}$ and hyperedges $h_j \in H$ (containing an arbitrary number of nodes), where the hyperedges are most generally heterogeneous. That is, we may wish to describe different types of hyperedges (e.g. bonds, triplets, and motifs) in the same hypergraph. These objects then have associated

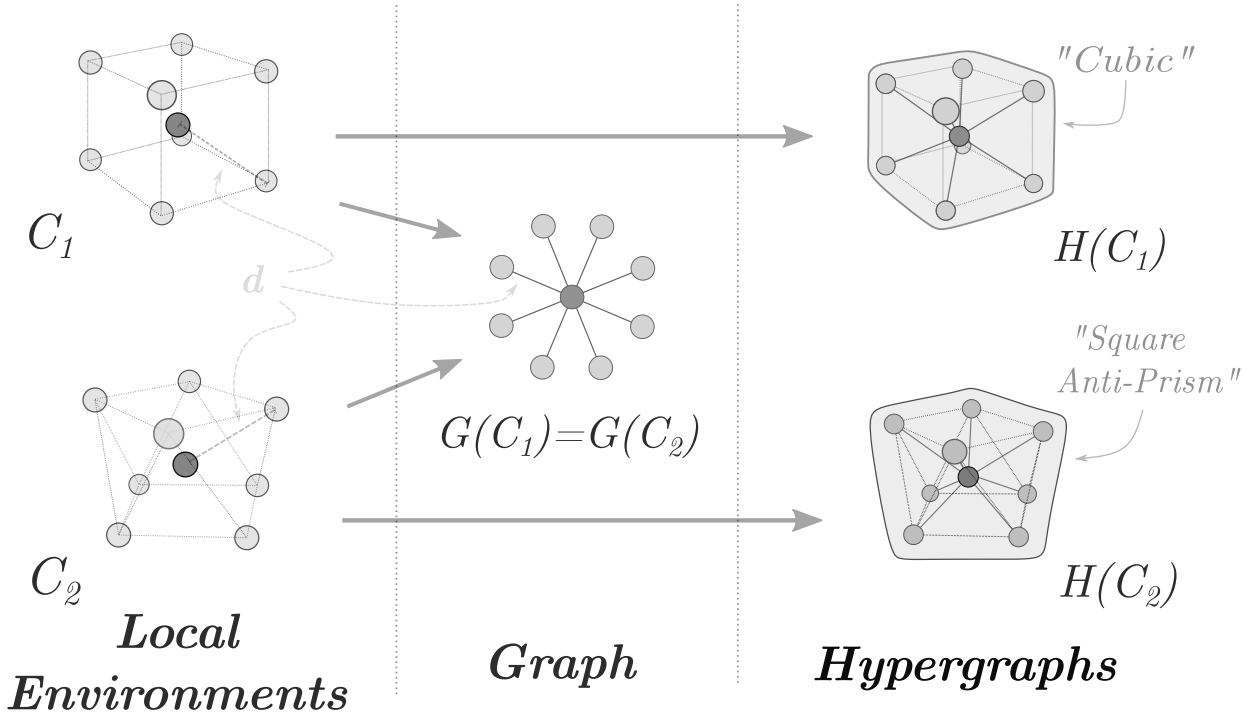


Figure 1: An example of two distinct geometries that are mapped to the same distance-based crystal graph. With inclusion of a first-shell feature vector encoding local geometry however, these structures are mapped to two distinct crystal hypergraphs. Note these are two possible coordination environments in oxides, determined statistically in [33].

feature vectors encoding relevant physical information, which we also refer to as v and h.

For the purpose of modeling material systems, we need to identify what different order structures are most important in their representation. Of course, atomic and bond level information is particularly important. However, higher-order structures may also be of interest, such as triplets of atoms and local environments of atoms, which we refer to as motifs in crystals.

Each of the aforementioned structures also has a natural set of distinct, coordinate-system invariant features that may be associated with them. At the triplet level (where two bonds share some common node), there is always a corresponding angle. While at the motif level, order parameters [39, 40] or continuous symmetry measures [23, 32] may be used to describe 3-dimensional coordination environments quantitatively. These different order structures may all be represented in a single crystal hypergraph.

Below, we discuss the generation of, and association of features with, all of the above-mentioned structures in crystalline solids.

Bond Edges

Bonds, or pair-wise atomic connections, are determined in the same manner as in a crystal graph. In the results below, we choose edges from a maximum number of neighbors $N_{max} = 12$ found within a shell of radius $r_{max} = 6\AA$.

Triplet Hyperedges

Triplet hyperedges are then formed from the set of bonds. For each set of bonds connected by one node, a triplet hyperedge is formed. The feature of these triplet hyperedges is also a Gaussian expansion, though now of the angle formed by the unit vectors of the two bonds [9]. Triplet hyperedges give us a way to incorporate some angular resolution into our representation scheme in a coordinate-system-invariant manner. For a node with N bonds then, there will be $N(N-1)/2$ triplets. Thus, the price we pay for complete angular resolution of any two bonds is a quadratic increase in the number of hyperedges.

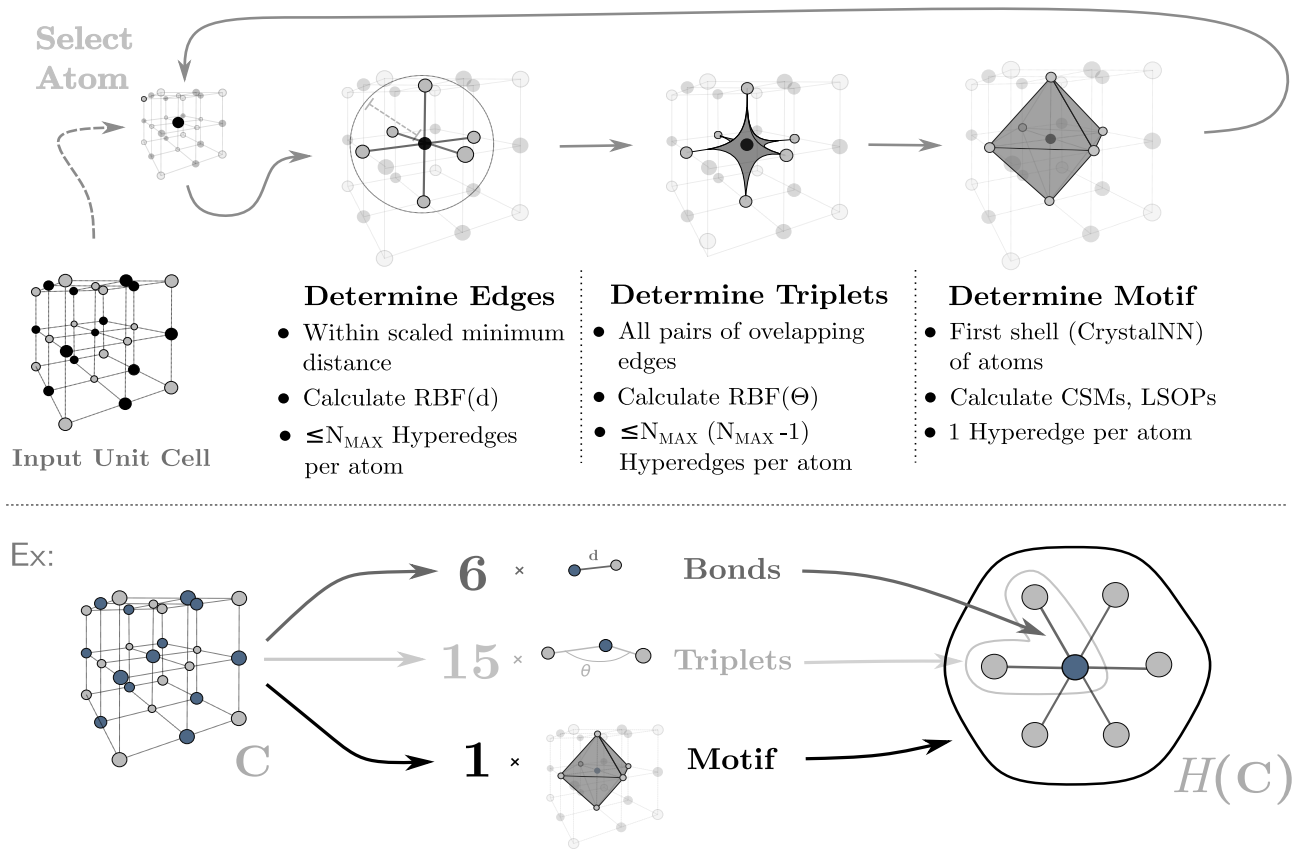


Figure 2: Typical construction loop for a crystal hypergraph. First, pair-wise bonds/edges are determined, then triplets are derived from overlapping pairs of bonds, and finally motifs are determined as first-shells of neighbors by some (generally more restrictive) criteria. Features for each and upper bounds on numbers of hyperedges for each type are also listed.

Motif Hyperedges

Motif determination may be achieved by a wide range of functions, and is akin to an algorithmic determination of coordination number [20]. Here, we use a modified Voronoi algorithm with a cut-off radius implemented as CrystalNN in pymatgen. Note this is a much stricter algorithm than that used to determine edges and triplets, since the motifs features depend heavily on the selected first-shell.

The features of these motifs are a concatenation of Zimmerman's 34 local structure order parameters [39, 40], and continuous symmetry measures [23] (e.g. 'distance to a perfect shape') for 59 common coordination environments. In essence, both are just sets of quantitative measures designed to describe 3 dimensional physical shape. Motifs give us a way to describe the local geometry of sites in material systems with much fewer hyperedges. Since each node will contribute one motif hyperedge, for a crystal with N_{nodes} nodes, we just have N_{nodes} motifs.

Crystal Hypergraph Convolution

We now must consider a message passing framework analogous to *Gilmore, et al* [15] but applying to hypergraph structures. That is, we now have:

$$m_v^{t+1} = \sum_{h_j \in \mathcal{N}(v)} M_t(n_v^t, h_j^t, \{n_w^t | n_w \in h_j\}),$$
$$n_v^{t+1} = U_t(n_v^t, m_v^{t+1}),$$
$$\hat{y} = R(\{n_v^T\}),$$

so that each node is still updated according to some layer-wise update function U_t, aggregating messages m^{t+1} formed from origin node features, hyperedge features h_j, and hyperedge neighborhood features $n_w \in h_j$. This update occurs node-wise

and then after T layers, some readout function R is used to output the corresponding predicted value $\hat{y}$, which utilizes the set of learned node features.

The biggest difference here is that we now need a message forming function M_t that accounts for a set of node features $\{n_w^t | n_w \in h_j\}$ which may vary in size between different hyperedges (even of the same type). This stands in opposition to the case of regular edges, where we are assured a fixed size of two nodes per edge.

One approach would be to fix the dimensionality of each type of hyperedge, or have a different convolutional operator for each different size hyperedge (as is effectively the approach taken with line graph networks [7]). Here, however, we wish to maintain generality in edge size so we need not fix hyperedge sizes for hyperedge type, since structures of different sizes may be described by similar metrics. For example, we may wish that motifs resembling polyhedra with different numbers of vertices are described by common sets of features.

Of course, there should be different message and update functions for each different order structure (bonds, triplets, motifs, etc.) with different features. This is accounted for by treating the data as a heterogeneous graph, with different hyperedge types. Below, we consider three strategies that allow us to apply our convolutional operator to hyperedges of arbitrary size.

Three Possible Approaches to Hyperedge Convolution

Three general approaches for message passing that account for this multi-order nature have been considered in this work: **1.** the construction of a hyperedge relatives graph, upon which regular graph convolution may be applied; **2.** total exchange hyperedge message convolution, which completely generalizes the CGCNN [35] and ALIGNN [9] models to hypergraphs; and **3.** neighborhood aggregation, which balances performance of the former approach by forming a single neighborhood feature for each hyperedge.

Each approach has a different computational cost in terms of the total number of messages, along with a potentially different practical definition of a hypergraph. These considerations are presented below, with a specific convolutional structure and empirical results on common test datasets presented after.

Hyperedge Relatives Graph

We may define a dual graph $\mathcal{D}(h)$ to a hypergraph h to be a graph in which nodes represent the hyperedges of the hypergraph, and connections represent the overlap of respective hyperedge neighborhoods. In the case of a crystal hypergraph with heterogeneous hyperedges, this dual graph is a graph with heterogeneous nodes. We term this heterogeneous dual graph of a crystal hypergraph the relatives graph for simplicity. Atomic features may be included in this framework by adding a singleton hyperedge for each node.

The definition of the relatives graph allows us to perform the usual methods of graph convolution on hyperedge features. Such an approach also allows us to define our relatives graph as we would a graph, with just a standard edge index.

However, this approach lacks the interaction of neighboring features in convolution via the connecting hyperedge. That is, without a clear definition of the edge attribute, messages are generally of the form below:

$$m_v^t = \sum_{h_j \in \mathcal{N}(v)} M_t(n_v^t, h_j^t)$$

in which we simply discard the neighborhood of other node features contained in the hyperedge.

Computationally, this approach has a total number of messages that scales linearly with average hyperedge size, since each hyperedge only contributes one message to each node it contains. Accounting only for node-hyperedge connections in a relatives graph derived from a hypergraph with m hyperedges of average order n, the total number of messages per convolution will scale as $\mathcal{O}(nm)$.

Total Exchange Message Passing

Of course, we may wish to incorporate the neighboring features of some representation via their connecting hyperedge. This may be accomplished by simply forming a message for every pair of connected representations along with their connecting hyperedge's representation.

$$m_v^t = \sum_{h_j \in \mathcal{N}(v)} \sum_{n_w \in h_j} M_t(n_v^t, h_j^t, n_w^t),$$

Here, though, we have introduced a new summation which may drastically increase the number of messages for larger hyperedges. In this scheme, if each hyperedge contains an average of n nodes and there are m hyperedges, the total number of messages exchanged per node-wise convolution will scale as $\mathcal{O}(n^2 m)$.

Neighborhood Aggregation

Since the number of messages will scale tremendously with larger hyperedges in the framework described above, we may seek a way to incorporate

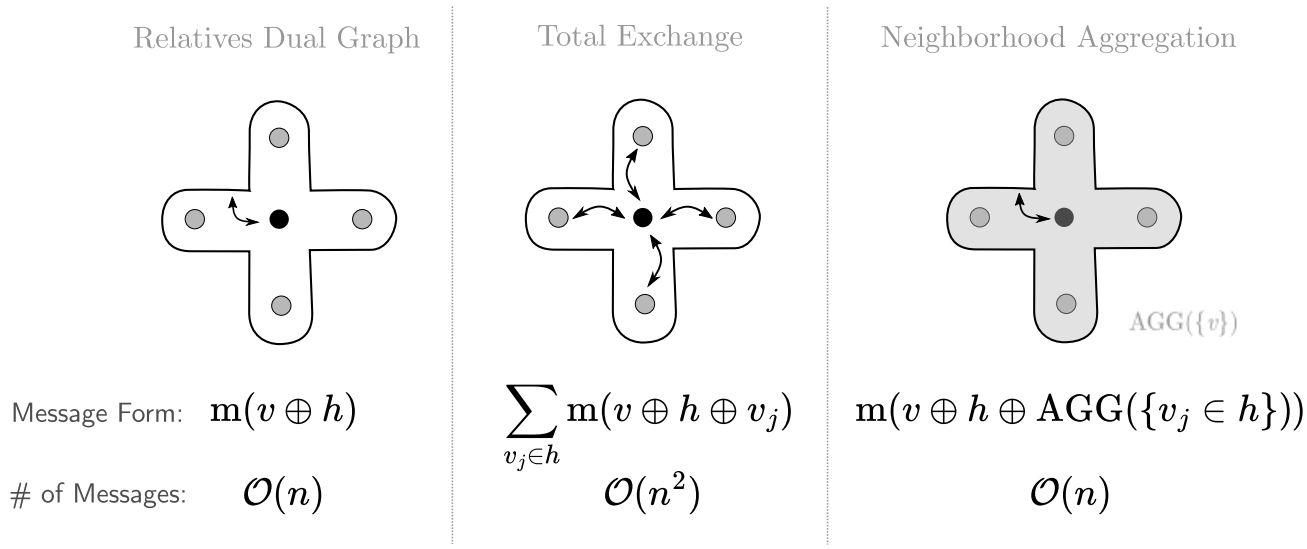


Figure 3: Overview of three possible message functions m for nodes v that generalize the message function in [15] to hyperedges h with more (or less) than two nodes. Here, n represents the average number of nodes in a hyperedge and the scaling relation is then for the total number of messages for one layer, for each hyperedge and each approach.

the neighborhood of features of a hyperedge into a single message. In this case, we may essentially form a 'neighborhood feature' representative of all contained nodes of a hyperedge. Typical aggregation methods may be used and trained to perform this neighborhood feature generation. Here then, we use the message functions of the form:

$$m_v^{t+1} = \sum_{h_j \in \mathcal{N}(v)} M_t(n_v^t, h_j^t, \mathrm{AGG}\big(\{n_w^t | n_w \in h_j\}\big)$$

This results in a set of node-wise messages that scales linearly with the average size of hyperedges, so that we now have a relationship of order $\mathcal{O}(nm)$ again, while still incorporating the features of neighboring nodes.

Model Architecture

In our CHGCNN model, initial atomic features were those used in CGCNN [35], consisting of a concatenated set of one-hot encoded atomic properties. For bond features, we used a Gaussian expansion of interatomic distance of dimension 40 ranging from 0 to 6 Å, triplet features were a Gaussian expansion of the cosine of bond angle, also of dimension 40, and motif features were a concatenation of 93 scalar order parameters and continuous symmetry measures. In the model considered in this work, initial node and hyperedge features were first passed into a linear embedding layer (with no activation function) with an output dimension of 64.

These embedded features were then fed into a set of Crystal HyperGraph Convolutional (CHGConv) layers which utilize the neighborhood aggregation method presented above. In CHGConv, we use a set of CGConv [35] layers applied to consecutively larger hyperedge types, taking as input the origin node of the hyperedge, the connecting hyperedge feature as the edge feature, and an aggregated set of neighborhood features as the connected node feature. So, for every CHGConv layer, the atoms are updated by each hyperedge type chosen once (see Fig. 4). Note that each CGConv for different hyperedge types have independent trainable parameters.

These learned node features are then mean pooled to form a crystal vector, which is passed to a fully connected layer and then projected down to a one-dimensional (scalar) output for regression. In the case of classification tasks, the fully connected layer, after mean pooling, utilized a dropout mechanism and output a probability distribution of classes by way of a softmax activation function.

Results & Discussion

Crystal hypergraph networks provide a unique opportunity to investigate the importance of different order structures in the prediction of various material properties. Specifically, we may compare performance between models based on differ-

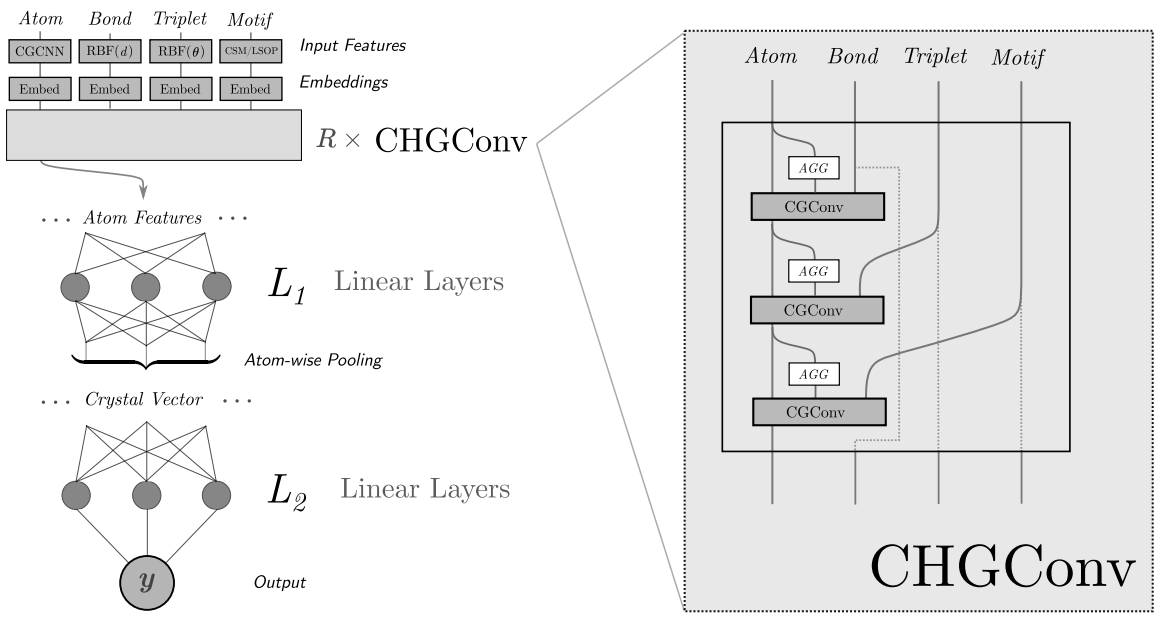


Figure 4: Example architecture for the crystal hypergraph convolutional network implemented in this work. Essentially, the model is a generalizaiton of CGCNN's [35] model architecture with CGConv being replaced by R hypergraph convolutional layers (CHGConv). Here, CHGConv updates nodes first according to edges (bonds), and then according to triplets or motifs.

ent types of hyperedges to probe the relevance of certain structures (e.g. motifs vs. triplets) in material property prediction. From the different hyperedge types considered here, we build three different models based on the architecture given in Fig. 4. We first test a basic bond-only network equivalent to CGCNN, and compare its performance against two models incorporating two types of hyperedges: bond-and-triplet and bond-and-motif models. For compound models (including more than one hyperedge type) each CHGConv layer performs convolution over the hyperedges in ascending order of hyperedge size. These models were each trained on sets of training data from two different databases of material properties: the Materials Project [16] and MatBench [12]. Details on the hyperparameters and training protocol can be found in the supplementary materials. Note that models using both motif and triplet-level edges, in general, diverged through training or did not perform any better than models using just motifs or triplets. As such, we only compare models using one or the other here.

We first focus on the comparative performance of models incorporating only bonds, models incorporating both bond-and-motif hyperedges and models with both bond-and-triplet hyperedges on five MatBench target datasets [12]. Mean absolute error (MAE) on validation sets for these tasks are reported in Table 1. These five datasets consist of the following targets: the highest frequency phonon peak ω_p for 1,265 materials, refractive indices n for 4,764 materials, formation energies E_f for a set of 18,829 perovskite materials, and 10,987 bulk and shear moduli, K_{vrh} and G_{vrh}, respectively. On all tasks of this set, the larger models incorporating both bonds and higher-order hyperedges performed best. Bond-and-triplet models performed best overall in predicting refractive indices and perovskite formation energies. For refractive index prediction, the triplet-based models had an average MAE of 0.398 across the test sets compared to an MAE of 0.424 for the bond-and-motif model, with both showing improvement over the bond-only model with an MAE of 0.454. Perovskite formation energy prediction results were very close for the three models, while the bond-and-triplet model again had the best performance with an MAE of 0.0598 eV/atom, the bond-and-motif model was close behind at an MAE of 0.0599 eV/atom, both showing only a slight improvement over the bond-only model's performance with an MAE of 0.604 eV/atom. However, since perovskites are a class of materials with a relatively standard structure, it should be unsurprising that the inclusion of additional structural information has little to no effect on performance.

Bond-and-motif models performed best in the remaining tasks. In the prediction of highest fre-

Hyperedge Types	ω_p (*1,265*) MAE (cm^{-1})	n (*4,764*) MAE	E_f (*18,829*) MAE (eV/Atom)	$\text{Log}_{10}(G_{vrh})$ (*10,987*) MAE (Log_{10}GPa)	$\text{Log}_{10}(K_{vrh})$ (*10,987*) MAE (Log_{10}GPa)
Bond-only	82.0	0.454	0.0604	0.1005	0.0816
Bond & Triplet	77.5	**0.398**	**0.0598**	0.0966	0.0757
Bond & Motif	**74.3**	0.424	0.0599	**0.0943**	**0.0730**

Table 1: Validation dataset results for five MatBench target sets: highest frequency phonon peak ω_p, refractive index n, perovskite formation energy E_f, and bulk moduli K_{vrh} and shear moduli G_{vrh}. Note that the italicized numbers below the target name correspond to the total size of each dataset. Best results are indicated in bold.

quency phonon peak, the bond-and-motif model had the best performance with an average MAE of 74.3 cm^{-1} across test sets compared to 77.5 cm^{-1} for the triplet based model and 82.1 cm^{-1} for the bond-only model. For elastic targets, motif based models boasted average MAEs of 0.0943 Log_{10}(GPa) and 0.0730 Log_{10}(GPa) on the test sets of bulk moduli G_{vrh} and shear moduli K_{vrh}, respectively. This is compared to the performance of the bond-and-triplet models with MAEs of 0.0966 Log_{10}(GPa) and 0.0757 Log_{10}(GPa) on G_{vrh} and K_{vrh}, with both being an improvement over the bond-only models with MAEs of 0.1005 Log_{10}(GPa) and 0.0816 Log_{10}(GPa). This may be indicative of the importance of such information in the relation of stress to infinitesimal strain, since the local environments of atoms would be of particular importance in considerations of shear response (that is, calculations of G_{vrh}) though, perhaps, less so in considerations of bulk response (K_{vrh}).

We now consider these models performance on three more target properties for a much larger set of 152,605 materials from the Materials Project database [16], with targets including formation energy E_f, band gap E_g, and metallicity. Results for this set of tests are reported in Table 2 with MAE reported for regression tasks and area under curve (AUC) for classification tasks. For the Materials Project dataset, the bond-and-triplet model performed best for all tasks. In the prediction of formation energy, the bond-and-motif model performed better than the bond-only model with an MAE of 0.0506 eV/atom vs. an MAE of 0.0512 eV/atom, while the motif-and-triplet model performed best with an MAE of 0.0484 eV/atom on the test set. This trend also held for the metal/nonmetal classification task, with the best performance on the test set again by the bond-and-triplet model with an AUC of 0.933. The bond-and-motif model had the next best performance on the test set, with an AUC of 0.932 compared to an AUC of 0.927 for the bond-only model. In the prediction of band gaps, the bond-and-motif model and bond-only model had prediction MAEs of 0.356 eV and 0.366 eV, respectively. The bond-and-triplet model again performed best with an MAE of 0.325 eV on the test set.

Perhaps the strongest point to be made in regard to these results is that for most tasks, motif information contributed to comparable or better performance than triplet-level results. This is at a significantly lower computational cost, in terms of the total number of messages exchanged through convolution, since the number of motifs is simply the number of atoms n, whereas the number of triplets scales with the average number of bonds per atom N as $N(N-1)/2$.

A similar observation was made in the AMDNet architecture [2], where motif information (included via an additional 'motif graph' for each material) also improved performance on most tasks, but here we compare results directly to the inclusion of bond angles via triplets. Our results indicate that one local neighborhood feature per atom may be sufficient to describe the local geometries of atoms for many predictive tasks, as opposed to the more data-intensive triplet representation scheme usually employed (often by way of line graphs). Taking this as a learned guiding principle, future crystal representations may benefit from reduced size while being assured similar geometric resolution. However, it should be noted that the greatest improvements seen here were on smaller datasets, suggesting the inductive bias of the motif-level hyperedges yields diminishing returns with larger datasets. This suggests that with large enough training data sets, continuous convolution filters for hyperedges encoding distance or angle alone may sufficient for larger targets. However, given the intrinsic difficulty of the production of most target data in the materials science, the specific methods presented here may still prove beneficial in many applications.

Conclusions

State-of-the-art GNN models applied to material property prediction often represent material sys-

Hyperedge Types	E_f Best MAE (eV/Atom)	E_g Best MAE (eV)	Metal/Non-metal Classification Best AUC
Bond-only	0.0512	0.356	.927
Bond & Triplet	**0.0484**	**0.325**	**.933**
Bond & Motif	0.0506	0.366	.932

Table 2: Validation dataset results for three Materials Project target sets: formation energy E_f, band gap E_g, and metalicity. Here, each dataset included a total of 152,605 materials after 300 epochs of training with 5-fold nested cross validation. Best results are indicated in bold.

tems as graphs with relatively low geometrical resolution. This low resolution is often increased by associating bond angles with auxiliary line graphs derived from the graph itself. The primary argument of our work is that hypergraphs are a more natural representation of material structures that allow us to explicitly incorporate geometrical information with different substructures of choice in one unified representation. The results suggest that such an approach allows for a substantial decrease in computational cost by incorporating such geometrical information with single local environment hyperedges for each node as opposed to triplets of atoms for each pair of overlapping bonds. This is shown within one unified framework to have comparable performance on a number of common predictive tasks.

Future works may investigate more powerful hypergraph convolutional operators that automatically detect motifs [38, 11, 28]; or apply this framework to molecular systems [10, 13] with functional groups. Inter-order convolution may also be of interest for certain tasks, where different hyperedge types may update each other's representations as opposed to just atom representations. Note that inter-order convolution would allow for a complete generalization of previous line-graph convolution schemes, where triplets effectively update their respective bonds' representations through convolution, as in [9, 24]. Other order structures (beyond motif-level) may also be of interest, such as hyperedges representing defect complexes or entire unit cells. Equivariant features and convolution [14, 29, 36] may also be incorporated for the prediction of coordinate-system dependent properties of materials from hypergraph representations, with the present work being focused on coordinate-system invariant features and targets.

Data Availability

The code used in this paper's results were built on pytorch-geometric and can be found in the following Github repository: https://github.com/qmatyanlab/CHGCNN. The processed data including bond, triplet, and motif features is available in a Zenodo repository with DOI: 10.5281/zenodo.14756640.

Acknowledgments

This work is supported by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, under Award No. DE-SC0023664. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231 using NERSC award BES-ERCAP0029544.

References

[1] Song Bai, Feihu Zhang, and Philip HS Torr. "Hypergraph convolution and hypergraph attention". In: *Pattern Recogn.* 110 (2021), article 107637.

[2] Huta R Banjade et al. "Structure motif–centric learning framework for inorganic crystalline systems". In: *Sci. Adv.* 7.17 (2021), article eabf1754.

[3] Jin Hyun Chang et al. "CLEASE: a versatile and user-friendly implementation of cluster expansion method". In: *J. Phys. Condens. Matter* 31.32 (2019), article 325901.

[4] Chen Chen et al. "Chemical environment adaptive learning for optical band gap prediction of doped graphitic carbon nitride nanosheets". In: *Neural Comput. Apl.* (2024), pp. 1–15. ISSN: 1433-3058. DOI: 10.1007/s00521-024-10775-1. URL: https://doi.org/10.1007/s00521-024-10775-1.

[5] Chi Chen and Shyue Ping Ong. "A universal graph deep learning interatomic potential for the periodic table". In: *Nat. Comput. Sci.* 2.11 (2022), pp. 718–728.

[6] Chi Chen et al. "Graph networks as a universal machine learning framework for molecules and crystals". In: *Chem. Mater.* 31.9 (2019), pp. 3564–3572.

[7] Zhengdao Chen, Xiang Li, and Joan Bruna. "Supervised community detection with line graph neural networks". In: *arXiv preprint arXiv:1705.08415* (2017).

[8] Jiucheng Cheng, Chunkai Zhang, and Lifeng Dong. "A geometric-information-enhanced crystal graph network for predicting properties of materials". In: *Commun. Mater.* 2.1 (2021), article 92.

[9] Kamal Choudhary and Brian DeCost. "Atomistic Line Graph Neural Network for improved materials property predictions". In: *npj Comput. Mater.* 7.1 (2021), pp. 1–8.

[10] Christopher R Collins et al. "Constant size descriptors for accurate machine learning models of molecular properties". In: *J. Chem. Phys.* 148 (2018), article 24.

[11] Jiadong Dan et al. "Exploring Motifs and Their Hierarchies in Crystals via Unsupervised Learning". In: *Microsc. Microanal.* 28.S1 (2022), pp. 3002–3003.

[12] Alexander Dunn et al. "Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm". In: *npj Comput. Mater.* 6.1 (Sept. 2020), pp. 138. ISSN: 2057-3960. DOI: 10.1038/s41524-020-00406-3. URL: https://doi.org/10.1038/s41524-020-00406-3.

[13] Nikita Fedik et al. "Extending machine learning beyond interatomic potentials for predicting molecular properties". In: *Nat. Rev. Chem.* 6.9 (2022), pp. 653–672.

[14] Mario Geiger and Tess Smidt. "e3nn: Euclidean neural networks". In: *arXiv preprint arXiv:2207.09453* (2022).

[15] Justin Gilmer et al. "Neural message passing for quantum chemistry". In: *34th Int. Conf. Mach. Learn.* Vol. 70. PMLR. 2017, pp. 1263–1272.

[16] Anubhav Jain et al. "Commentary: The Materials Project: A materials genome approach to accelerating materials innovation". In: *APL Mater.* 1.1 (July 2013), article 011002. ISSN: 2166-532X. DOI: 10.1063/1.4812323. eprint: https://pubs.aip.org/aip/apm/article-pdf/doi/10.1063/1.4812323/13163869/011002_1_online.pdf. URL: https://doi.org/10.1063/1.4812323.

[17] RB King. "The shapes of coordination polyhedra". In: *J. Chem. Ed.* 73.10 (1996), article 993.

[18] Yingli Liu et al. "Machine learning in materials genome initiative: A review". In: *J. Mater. Sci. Tech.* 57 (2020), pp. 113–122. ISSN: 1005-0302. DOI: https://doi.org/10.1016/j.jmst.2020.01.067. URL: https://www.sciencedirect.com/science/article/pii/S1005030220303327.

[19] Md Hosne Mobarak et al. "Scope of machine learning in materials research—A review". In: *Appl. Surf. Sci. Adv.* 18 (2023), article 100523. ISSN: 2666-5239. DOI: https://doi.org/10.1016/j.apsadv.2023.100523. URL: https://www.sciencedirect.com/science/article/pii/S2666523923001575.

[20] Hillary Pan et al. "Benchmarking Coordination Number Prediction Algorithms on Inorganic Crystal Structures". In: *Inorg. Chem.* 60.3 (Feb. 2021), pp.1590–1603. ISSN: 0020-1669. DOI: 10.1021/acs.inorgchem.0c02996. URL: https://doi.org/10.1021/acs.inorgchem.0c02996.

[21] Cheol Woo Park and Chris Wolverton. "Developing an improved crystal graph convolutional neural network framework for accelerated materials discovery". In: *Phys. Rev. Mater.* 4.6 (2020), article 063801.

[22] Linus Pauling. "THE PRINCIPLES DETERMINING THE STRUCTURE OF COMPLEX IONIC CRYSTALS". In: *J. Chem. Soc.* 51.4 (Apr. 1929), pp. 1010–1026. ISSN: 0002-7863. DOI: 10.1021/ja01379a006. URL: https://doi.org/10.1021/ja01379a006.

[23] Mark Pinsky and David Avnir. "Continuous symmetry measures. 5. The classical polyhedra". In: *Inorg. Chem.* 37.21 (1998), pp. 5575–5582.

[24] Robin Ruff et al. "Connectivity optimized nested line graph networks for crystal structures". In: *Digit. Discov.* 3.3 (2024), pp. 594–601.

[25] J.M. Sanchez, F. Ducastelle, and D. Gratias. "Generalized cluster description of multicomponent systems". In: *Physica A Stat. Mech. Appl.* 128.1 (1984), pp. 334–350. ISSN: 0378-4371. DOI: https://doi.org/10.1016/0378-4371(84)90096-7. URL: https://www.sciencedirect.com/science/article/pii/0378437184900967.

[26] Erik E Santiso and Bernhardt L Trout. "A general set of order parameters for molecular crystals". In: *J. Chem. Phys.* 134 (2011), article 6.

[27] K. T. Schütt et al. "SchNet – A deep learning architecture for molecules and materials". In: *J. Chem. Phys.* 148.24 (Mar. 2018), article 241722. ISSN: 0021-9606. DOI: 10.1063/1.5019779. eprint: https://pubs.aip.org/aip/jcp/article-pdf/doi/10.1063/1.5019779/16655678/241722_1_online.pdf. URL: https://doi.org/10.1063/1.5019779.

[28] Killian Sheriff, Yifan Cao, and Rodrigo Freitas. "Chemical-motif characterization of short-range order with E(3)-equivariant graph neural networks". In: *npj Comput. Mater.* 10.1 (Sept. 2024), article 215. ISSN: 2057-3960. DOI: 10.1038/s41524-024-01393-5. URL: https://doi.org/10.1038/s41524-024-01393-5.

[29] Nathaniel Thomas et al. "Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds". In: *arXiv preprint arXiv:1802.08219* (2018). arXiv: 1802.08219 [cs.LG]. URL: https://arxiv.org/abs/1802.08219.

[30] A. van de Walle. "A complete representation of structure–property relationships in crystals". In: *Nat. Mater.* 7.6 (June 2008), pp. 455–458. ISSN: 1476-4660. DOI: 10.1038/nmat2200. URL: https://doi.org/10.1038/nmat2200.

[31] Logan Ward et al. "A general-purpose machine learning framework for predicting properties of inorganic materials". In: *npj Comput. Mater.* 2.1 (2016), pp. 1–7.

[32] David Waroquiers et al. "ChemEnv: a fast and robust coordination environment identification tool". In: *Acta Crystallogr. B* 76.4 (2020), pp. 683–695.

[33] David Waroquiers et al. "Statistical analysis of coordination environments in oxides". In: *Chem. Mater.* 29.19 (2017), pp. 8346–8360.

[34] Jing Wei et al. "Machine learning in materials science". In: *InfoMat* 1.3 (2019), pp. 338–358.

[35] Tian Xie and Jeffrey C Grossman. "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties". In: *PRL* 120.14 (2018), article 145301.

[36] Keqiang Yan et al. "Complete and Efficient Graph Transformers for Crystal Material Property Prediction". In: *arXiv preprint arXiv:2403.11857* (2024). arXiv: 2403.11857 [cs.LG]. URL: https://arxiv.org/abs/2403.11857.

[37] Zhu Yang and Lei-Han Tang. "Coordination motifs and large-scale structural organization in atomic clusters". In: *Phys. Rev. B* 79 (4 Jan. 2009), article 045402. DOI: 10.1103/PhysRevB.79.045402. URL: https://link.aps.org/doi/10.1103/PhysRevB.79.045402.

[38] Shichang Zhang et al. "Motif-driven contrastive learning of graph representations". In: *arXiv preprint arXiv:2012.12533* (2020).

[39] Nils E. R. Zimmermann et al. "Assessing Local Structure Motifs Using Order Parameters for Motif Recognition, Interstitial Identification, and Diffusion Path Characterization". In: *Front. Mater.* 4 (2017), article 34. ISSN: 2296-8016. DOI: 10.3389/fmats.2017.00034. URL: https://www.frontiersin.org/articles/10.3389/fmats.2017.00034.

[40] Nils ER Zimmermann and Anubhav Jain. "Local structure order parameters and site fingerprints for quantification of coordination environment and crystal structure similarity". In: *RSC Adv.* 10.10 (2020), pp. 6063–6081.

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. The list of order parameters included those implemented in pymatgen code and described in [39, 40].

A CSM is defined precisely so that it may act as a 'distance' from some prototypical shape to some given structure.

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.

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

In the model utilized in this work, we generally employed use of a learnable set of common aggregation functions for the neighborhood feature aggregation (AGG above), inspired by *ChemGNN* [4].

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, motif).

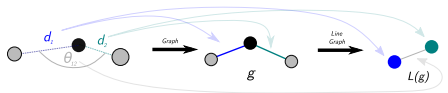
Stochastic gradient descent (SGD) was used as an optimizer through training with an initial learning rate of 0.01. A multi-step learning rate scheduler divided this learning rate by a factor of 10 at epoch 150, with training running for a total of 300 epochs.

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 MSE (for regression tasks) and cross entropy (for classification tasks). Accuracy is then reported in 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.

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 [9, 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.

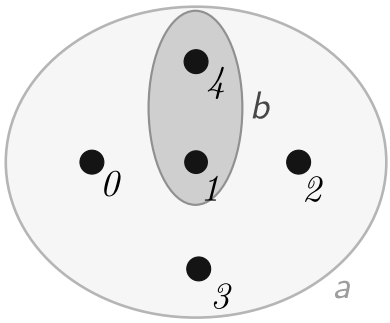
Our argument against such representation schemes here is that the order of messages grows combinatorically for derived line graphs as $\mathcal{O}(nm^2)$, where n is the number of nodes and m is the average number of edges per node in $\mathcal{G}$.

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.

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 [1]).



Hyperedge Index: [[*0,1,2,3,4,1,4*], [a,a,a,a,a,b,b]]