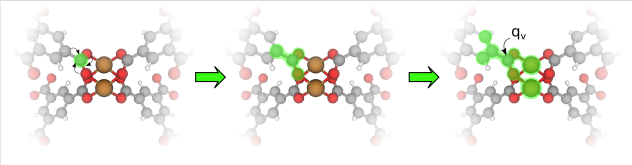
Message passing neural networks for partial charge assignment to metal-organic frameworks

Ali Raza,[†] Arni Sturluson,[‡] Cory Simon,[*,‡] and Xiaoli Fern[*,†]

[†]*School of Electrical Engineering and Computer Science. Oregon State University.*
[‡]*School of Chemical, Biological, and Environmental Engineering. Oregon State University.*

E-mail: Cory.Simon@oregonstate.edu; xfern@eecs.oregonstate.edu



Abstract

Virtual screenings can accelerate and reduce the cost of discovering metal-organic frameworks (MOFs) for their applications in gas storage, separation, and sensing. In molecular simulations of gas adsorption/diffusion in MOFs, the adsorbate-MOF electrostatic interaction is typically modeled by placing partial point charges on the atoms of the MOF. For the virtual screening of large libraries of MOFs, it is critical to develop computationally inexpensive methods to assign atomic partial charges to MOFs that accurately reproduce the electrostatic potential in their pores. Herein, we design and train a message passing neural network (MPNN) to predict the atomic partial charges on MOFs under a charge neutral constraint. A set of ca. 2 250 MOFs labeled with high-fidelity partial charges, derived from periodic electronic structure calculations, serves as training examples. In an end-to-end manner, from charge-labeled crystal graphs representing MOFs, our MPNN machine-learns features of the local bonding environments of the atoms and learns to predict partial atomic charges from these features. Our trained MPNN assigns high-fidelity partial point charges to MOFs with orders of magnitude lower computational cost than electronic structure calculations. To enhance the accuracy of virtual screenings of large libraries of MOFs for their adsorption-based applications, we make our trained MPNN model and MPNN-charge-assigned computation-ready, experimental MOF structures publicly available.

1 Introduction

Metal-organic frameworks (MOFs) are nanoporous materials that often exhibit large internal surface areas.[1] Because MOFs selectively adsorb gas into their pores/on their internal surface, MOFs can be used to store,[2] separate/purify,[3] and sense[4] gases. Moreover, MOF structures are highly adjustable and therefore can be tuned to optimize a target adsorption property for a given adsorption-based engineering application.[5] The adjustability of MOFs stems from their modular synthesis: metals/metal clusters and organic linker molecules self-assemble into a crystalline structure.[1] By changing the molecular building blocks, many MOFs with diverse pore shapes and internal surface chemistries can be synthesized; on the order of 10 000 porous MOFs[6,7] have been reported to date.

Molecular models and simulations play an important role in the discovery and deployment of MOFs for adsorption-based applica-

tions.[8–10] Instead of an Edisonian approach to find MOFs that meet a target adsorption property, classical molecular models and simulations can quickly and cost-effectively predict the adsorption properties of thousands of MOFs and shortlist the most promising subset for experimental investigation. High-throughput computational screenings of MOFs have directly led to the discovery of high-performing MOFs for carbon dioxide capture,[11,12] xenon/krypton separations,[13] oxygen storage,[14] hydrogen storage,[12] and mustard gas capture.[15] In addition to virtual screening, molecular simulations of gas adsorption in MOFs can elucidate the most favorable adsorption sites in a MOF,[16,17] explain anomalous adsorption phenomena,[18,19] and uncover structure-property relationships.[20,21]

The molecular mechanics description of adsorbate-MOF interactions* used in a molecular simulation typically consists of the sum of a van der Waals and an electrostatic contribution.[10] The electrostatic interaction is particularly important for adsorbates with polar bonds, such as CO_2 and H_2O.[22–24] To model the adsorbate-MOF electrostatic interaction, we must model the electrostatic potential in the pores of the MOF, created by the atoms of the MOF. Typically, the MOF-hosted electrostatic potential is described by placing (fixed) partial (i.e. non-integer) point charges at the centers of the atoms of the MOF.[25] Molecular models for adsorbate molecules, such as CO_2 and N_2,[26] also possess partial point charges, which, during a molecular simulation, interact with the point charges on the MOF via Coulomb's law to comprise the adsorbate-MOF electrostatic potential energy of interaction†. There are several methods to assign partial charges to the atoms of a MOF,[25] which are non-observable. Choosing a charge assignment method for the virtual screening of a large library of MOFs often involves a trade-off between computational cost and accuracy of the representations of the electrostatic potential in the pores of the MOFs. Notably, the simulated adsorption properties and thus ranking of MOFs in virtual screenings could be highly dependent on the (accuracy of the) charge assignment method.[22,28–30]

Broadly, methods to assign partial point charges to a MOF[25] take two different approaches: (1) use an electronic structure calculation (e.g. a density functional theory (DFT) calculation) to obtain the electrostatic potential/electron density (a) in the pores of the (periodic) MOF or (b) surrounding a non-periodic cluster representation of the MOF, then derive charges that are consistent with this electrostatic potential/electron density; (2) assign charges using a (semi)empirical model whose parameters were fit to experimental data or to charges assigned by approach (1). Approach (1) includes Repeating Electrostatic Potential Extracted Atomic (REPEAT) charges[31] and Density Derived Electrostatic and Chemical (DDEC)[32] charges. In molecular building block-based charge assignment, MOFs inherit the charges of their molecular building blocks,[33] so that molecular charges, derived from electronic structure calculations, on a set of linker molecules and a set of metal clusters provide charges for a combinatorial number of MOFs. Approach (2) includes charge equilibration methods (QEq),[34,35] statistical machine learning models,[36–39] and nearest-neighbor-like approaches based on the chemical element and bonding environment of the atom.[40–43] Generally, approach (1) produces a more accurate electrostatic potential in the pores of the MOF but incurs a computational cost orders of magnitude greater than the cost of approach (2). Thankfully, Nazarian et al.[44] performed periodic DFT calculations to obtain the electron densities in ca. 2 900 experimentally synthesized MOFs[45] and assigned chemically meaningful, high-quality partial point charges to each MOF via the DDEC method.[32] Still, a large number of MOFs lack high-quality charges: (i) the majority of the second version (v2) of the computation-ready, experimental (CoRE) MOF dataset[7] of ca. 14 000 structures;

*We focus on classical as opposed to quantum methods to describe adsorbate-MOF interactions because quantum methods are too computationally costly to be used for molecular simulations in thousands of MOFs.

†This electrostatic potential energy of interaction is usually computed via the Ewald summation[27] given that MOFs are modeled as periodic systems.

(ii) newly synthesized MOFs that are continually reported;[6] and (iii) libraries of hypothetical/predicted MOF crystal structure models constructed with the aim of discovering new MOFs that have not been synthesized in the laboratory.[46–48] Screening these MOFs for gas storage, separations, and sensing via molecular simulations demands a computationally cheap *and* high-fidelity method for MOF charge assignment.

In this work, we develop and train a message passing neural network (MPNN)[49,50] architecture to assign partial point charges to each atom of a MOF structure under a charge-neutral constraint. To enable our machine-learning of high-fidelity charges on MOFs, we leverage the database of DFT-derived partial point charges on ca. 2 900 experimentally synthesized MOFs by Nazarian et al.[44] as training examples. Our fundamental hypothesis, supported by Refs. 41,43, is that the charge of any given atom in a MOF is primarily determined by its chemical identity and local bonding environment. As opposed to manually engineering a feature to represent the local bonding environments of atoms,[51,52] we allow the MPNN to machine-learn vector representations of local bonding environments within MOFs. From the machine-learned features of the local bonding environments of the atoms, the MPNN then predicts their partial point charges. We train the MPNN to do this in an end-to-end manner, making the features of local bonding environments dense with information predictive of partial charge. The MOF crystal structures, represented as undirected graphs (nodes: atoms, edges: bonds) with node features encoding their chemical identities, are the direct inputs to the MPNN. Edges (bonds) across the unit cell boundary are included to account for periodicity. The MPNN sequentially passes information along the edges of the graph to learn/construct the vector representations of the local bonding environments. We enforce charge neutrality on a MOF by modeling the probabilistic distribution of charge on an atom within its local bonding environment and invoking the maximum likelihood principle under a charge neutral constraint. This allows the MPNN to give more slack to the charge of atoms with high variance when enforcing charge neutrality. Interestingly, our MPNN begins with an embedding layer that learns an information-dense representation of the chemical elements, encoding their typical charge.

Our trained MPNN assigns high-fidelity (treating the DFT-derived DDEC charges[44] as ground truth) charges to MOF atoms (mean absolute deviation on test MOFs, 0.025), outperforming a suite of charge equilibration methods[35] (minimum mean absolute deviation, 0.118, by I-QEq[53]), while incurring orders of magnitude lower computational cost than electronic structure calculations. To enable accurate virtual screenings of large libraries of MOFs for their adsorption-based applications, we make our trained MPNN model available to the molecular simulation community for MOF charge assignment and provide `.cif` files of MPNN-charge-assigned v2 computation-ready, experimental MOFs.[7]

2 Review of previous work

Ref. 25 reviews methods for assigning atomic partial charges to MOFs to enable molecular simulation of gas adsorption and diffusion. The most accurate, but computationally costly approach is to use an electronic structure calculation (e.g. DFT) to obtain the periodic electrostatic potential/electron density in the pores of the MOF, then derive point charges that are consistent with this (e.g., REPEAT[31] and DDEC[32])‡. The less accurate,

‡Consider the case where we use an electronic structure calculation to obtain the periodic electrostatic potential at a 3D grid of points superimposing the unit cell of the MOF. Instead of translating this grid into a set of partial point charges that can reproduce it, directly interpolating this grid during a molecular simulation of adsorption[28] confers both (i) higher accuracy, as there may be model error in the translation of the grid into a set of partial point charges and (ii) greater speed, as using Ewald summations to compute the electrostatic potential created by the point charges is likely more computationally expensive than grid interpolation. Counter arguments are that (i) storing the 3D electrostatic potential grid will consume more disk space and memory during the simulation and (ii) if the MOF is flexible

but computationally cheap approach is to use a (semi)empirical model to assign charges to MOF atoms, whose parameters were fit to experimental data or charges assigned with electronic structure calculations as input. Semiempirical charge equilibration (QEq) methods[34] are commonly used to assign point charges to MOFs owing to their low computational cost. Ongari et al.[35] review and compare several QEq variants and assess their correlation with the DFT-derived DDEC charges of ca. 2 900 MOFs by Nazarian et al.[44] The ionizing (I)-QEq[53] variant produced charges closest to the DDEC charges (mean absolute deviation 0.118), but there were significant deviations, which then propagate onto e.g. carbon dioxide adsorption in a molecular simulation.

Along the direction of this work, a few authors trained supervised machine learning models to assign partial charges to atoms of molecules (not periodic MOFs), using descriptors/fingerprints of the local environment of an atom that are either (i) manually engineered[36–38] or (ii) learned end-to-end by a message passing neural network.[39] An interesting subproblem is that, when a supervised model predicts the charge of each atom in the molecule pseudo-independently, based on its local environment, charge neutrality of the molecule is not guaranteed. To enforce charge neutrality, directly after the model assigns (preliminary) charges to the atoms, Refs. 36,37 distributed the negative of the excess charge of the molecule among its atoms, not uniformly, but based on the variances of the predicted charges the atoms by an ensemble of decision tree regressors. Atoms associated with more (less) variance received more (less) of the excess charge. Ref. 43 distributed excess charge based on the magnitude of the predicted charge. Ref. 39 enforced charge neutrality by predicting the electronegativity and hardness of an atom in its local bonding environment instead of directly predicting its charge, then minimizing the potential energy of the atoms of the molecule under a charge neutrality constraint.

during the simulation, the (fixed) partial point charges that follow the atoms can still be used, whereas the grid cannot.

Using the molecular graph as direct input, MPNNs have recently been employed to predict several different properties of molecules,[50,54–62] including antibacterial efficacy, DFT-calculated properties, solubility, photovoltaic efficiency, odor, and drug efficacy. The key advantage/novelty of MPNNs is that, instead of manually engineering molecular descriptors,[51,52] the MPNN automatically learns a (task-specific) descriptor of the molecule and its set of local bonding environments from the molecular graph in an end-to-end manner, while being trained to perform a prediction task.[54] In contrast to mapping a molecule (represented as a graph) to a single property, our MPNN architecture is unique in that it maps a crystal (represented as an undirected graph with edges across the periodic boundary included) to targets (partial charges) on each node (atom) under a graph-level (charge neutrality) constraint.

3 Problem formulation

Here, we mathematically formulate the partial charge assignment problem. Notation is listed in Tab. 1. We use bold lowercase letters for vectors and bold uppercase letters for matrices.

We represent the crystal structure of each MOF as an undirected graph with node features, $G = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, where $\mathcal{V}$ is the set of $n_v = |\mathcal{V}|$ nodes or vertices, representing atoms, $\mathcal{E}$ is the set of edges, representing bonds, and $\mathbf{X} \in \mathbb{R}^{d \times n_v}$ is the node feature matrix. The feature vector of node (atom) v in the graph (MOF), $\mathbf{x}_v \in \mathbb{R}^d$, is a one-hot encoding of its chemical element and is column v of $\mathbf{X}$. Let $e_{uv} = \{u, v\} \in \mathcal{E}$ denote an edge (bond) between nodes (atoms) u and v. Let the adjacency matrix of the graph be $\mathbf{A} \in \mathbb{R}^{n_v \times n_v}$, where $A_{uv} = 1$ if nodes u and v are connected by an edge and $A_{uv} = 0$ otherwise. Together, the adjacency matrix $\mathbf{A}$ and node feature matrix $\mathbf{X}$ characterize the crystal graph of a MOF.

Our goal is to learn a function $\mathbf{f}$ that takes the crystal graph G as input and outputs a predicted charge on each node:

$$(\mathbf{X}, \mathbf{A}) \mapsto \mathbf{f}(\mathbf{X}, \mathbf{A}) = \mathbf{q}, \tag{1}$$

Table 1: Notation and definitions.

symbol	description
G	the graph
$\mathcal{V}$	set of nodes in graph
$\mathcal{E}$	set of edges in graph
n_v	number of nodes in graph
n_e	number of edges in graph
e_{ij}	edge between nodes i and j
$\mathbf{A} \in \mathbb{R}^{n_v \times n_v}$	adjacency matrix of graph
$\mathbf{x}_v \in \mathbb{R}^d$	feature vector of node v (one-hot encoding of element)
$\mathbf{X} \in \mathbb{R}^{d \times n_v}$	feature matrix of graph
$\mathbf{x}_v^e \in \mathbb{R}^r$	element embedding of node v
$\mathbf{X}^e \in \mathbb{R}^{r \times n_v}$	node embedding matrix of graph
t	message passing time
$\mathbf{h}_v^{(t)} \in \mathbb{R}^k$	hidden representation of node v after t messages
$\mathbf{H}^{(t)} \in \mathbb{R}^{k \times n_v}$	hidden node feature matrix of graph after t messages
q_v	partial charge on node v (units: electron charge)
$\hat{q}_v$	predicted partial charge on node v (units: electron charge)
$\mathbf{q} \in \mathbb{R}^{n_v}$	charge vector of graph
$\mathcal{N}(v)$	neighbors of node v
$\boldsymbol{\beta}_\square$ ($\boldsymbol{B}_\square$)	vector (matrix) of weights
$[\mathbf{a}, \mathbf{b}]$	concatenation of vectors $\mathbf{a}$ and $\mathbf{b}$

while satisfying the charge neutrality constraint:

$$\sum_{v=1}^{n_v} q_v = 0. \tag{2}$$

Here, q_v is the charge on node v and element v of the charge vector $\mathbf{q} \in \mathbb{R}^{n_v}$ of the graph. The function $\mathbf{f}$ will be equivariant (i.e., $\mathbf{f}(\mathbf{XP}, \mathbf{PAP}^\intercal) = \mathbf{Pq}$ where $\mathbf{P}$ is a permutation matrix that permutes the nodes) so that the ordering of the atoms is immaterial.

4 Machine-learning partial charges

4.1 Converting a MOF crystal structure to a graph

We first describe how we convert a MOF crystal structure stored in a `.cif` file into an undirected graph $G = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ (nodes/atoms: $\mathcal{V}$, edges/bonds: $\mathcal{E}$, node features encoding chemical elements: $\mathbf{X}$). A `.cif` file of a MOF provides its unit cell vectors and a list of its atoms and their coordinates.

Nodes and Edges. For each atom in the unit cell of the MOF, we create a node to represent it. To build the adjacency matrix, $\mathbf{A}$, we must automatically infer which atoms are bonded based on their identities and atomic coordinates. We used a bond assignment algorithm from Refs. 58,63 that considers both the typical bond lengths of atoms as well as the arrangements of nearby atoms. We assign an edge (bond) between nodes u and v if (1) the periodic Euclidean distance between them is less than the sum of their covalent radii[64] plus a 0.25 Å tolerance and (2) they share a Voronoi face. By applying the minimum image convention when computing the distance, we include edges between atoms bonded across the periodic boundary. To determine which atoms share a Voroni face with atom u, we used Scipy[65] to compute the Voronoi diagram of all atoms within a 6 Å radius of atom u, periodic images included. To ensure bonds were properly formed with metal atoms commonly found in MOFs, we increased the covalent radius for ten metals (see Sec. S2).

Node features. For each node v, we construct its feature vector, $\mathbf{x}_v \in \mathbb{R}^d$, as a one-hot encoding of its chemical element. i.e., entry i of $\mathbf{x}_v$ is one if atom v in the MOF is chemical element i and zero otherwise. Among the charge-labeled MOFs,[44] there were 74 unique chemical elements (see Tab. S6), so $d = 74$. The node feature vectors comprise the columns of the node feature matrix $\mathbf{X}$ of the MOF.

Target vector. We construct the charge vector $\mathbf{q}$ for the DFT-derived, DDEC charge-labeled MOFs[44] whose `.cif` files contain a col-

umn with the partial atomic charge on each atom.

Ordering, equivariance, rotation- and translation-invariance. Of course, element v of the target (charge) vector $\mathbf{q}$, column v of the node feature matrix $\mathbf{X}$, and row/column v of the adjacency matrix $\mathbf{A}$ all represent the same atom in the MOF. However, the ordering of atoms in the crystal structure file is immaterial, as the function $\mathbf{f}$ in eqn. 1 learned by the MPNN is equivariant[49] to permutations of the nodes. Notably, the graph representation of the MOF is also rotation- and translation-invariant.

4.2 Neural architecture

Fig. 1 shows the architecture of our message passing neural network (MPNN)[49] to assign partial point charges to each node of a graph, representing a MOF crystal structure, under a charge neutral constraint. This MPNN, described in detail below, constitutes the function $\mathbf{f}$ in eqn. 1 that obeys the constraint in eqn. 2. Our MPNN architecture is composed of, sequentially, (1) an element embedding layer to map the node features (the one-hot encodings of chemical elements) into information-dense chemical element representations for initializing hidden node features, (2) a gated graph neural network[66] that passes messages between neighboring nodes, along the edges of the graph, to learn/construct hidden node representations that encode the local bonding environment of each node, and (3) node-level charge prediction under the graph-level charge neutrality constraint.

4.2.1 Chemical element embedding

First, we map each node feature vector $\mathbf{x}_v$ to a compressed representation, $\mathbf{x}_v^e \in \mathbb{R}^r$ ($r << d$):

$$\mathbf{x}_v^e = \texttt{sigmoid}.(\boldsymbol{B}_e \mathbf{x}_v). \tag{3}$$

The learned matrix of weights $\boldsymbol{B}_e$ is shared across all nodes. Because each node feature $\mathbf{x}_v$ is a one-hot encoding of a chemical element, the *embeddings* of the chemical elements are the columns of the matrix $\boldsymbol{B}_e$ passed through a sigmoid activation function (. for element-wise) to limit the range. Thus, the element embedding layer in eqn. 3 maps each chemical element, one-hot encoded in $\mathbf{x}_v$, to a low-dimensional, dense feature vector that encodes its typical charge. The hidden feature vector of node v is initialized using its element embedding $\mathbf{x}_v^e$ to facilitate training in the message passing phase, which we describe next.

4.2.2 Message passing

In the message passing phase,[50] a gated graph neural network (GGNN)[66] iteratively updates the hidden features (representations) of the nodes by passing information between neighboring nodes, along the edges of the graph. The *message* received by a node is a conglomeration of the information received from its neighbors. The GGNN employs a gated recurrent unit (GRU)[67] to, at each time step, update the hidden representation of each node using its current hidden representation and the message from its neighbors. We perform message passing for T time steps; at the end, each hidden node feature encodes the local bonding environment, which we define more precisely below, of the atom it represents.

The initial hidden representation of node v, $\mathbf{h}_v^{(0)} \in \mathbb{R}^k$, is set as its chemical element embedding concatenated with a zero vector:

$$\mathbf{h}_v^{(0)} = [\mathbf{x}_v^e, \mathbf{0}]. \tag{4}$$

We concatenate with the zero vector of dimensionality $k - r$ to allow the hidden representation to be higher-dimensional than the element embedding; conceptually, this is to account for the higher information content in the hidden node representation than in the element embedding of the node, as the former encapsulates both the atomic species of the node (as does the element embedding) *and* the surrounding bonding environment of the node (which the element embedding does not).

In taking a message passing time step from time t to $t + 1$, each node collects, sums, and transforms the hidden representations of

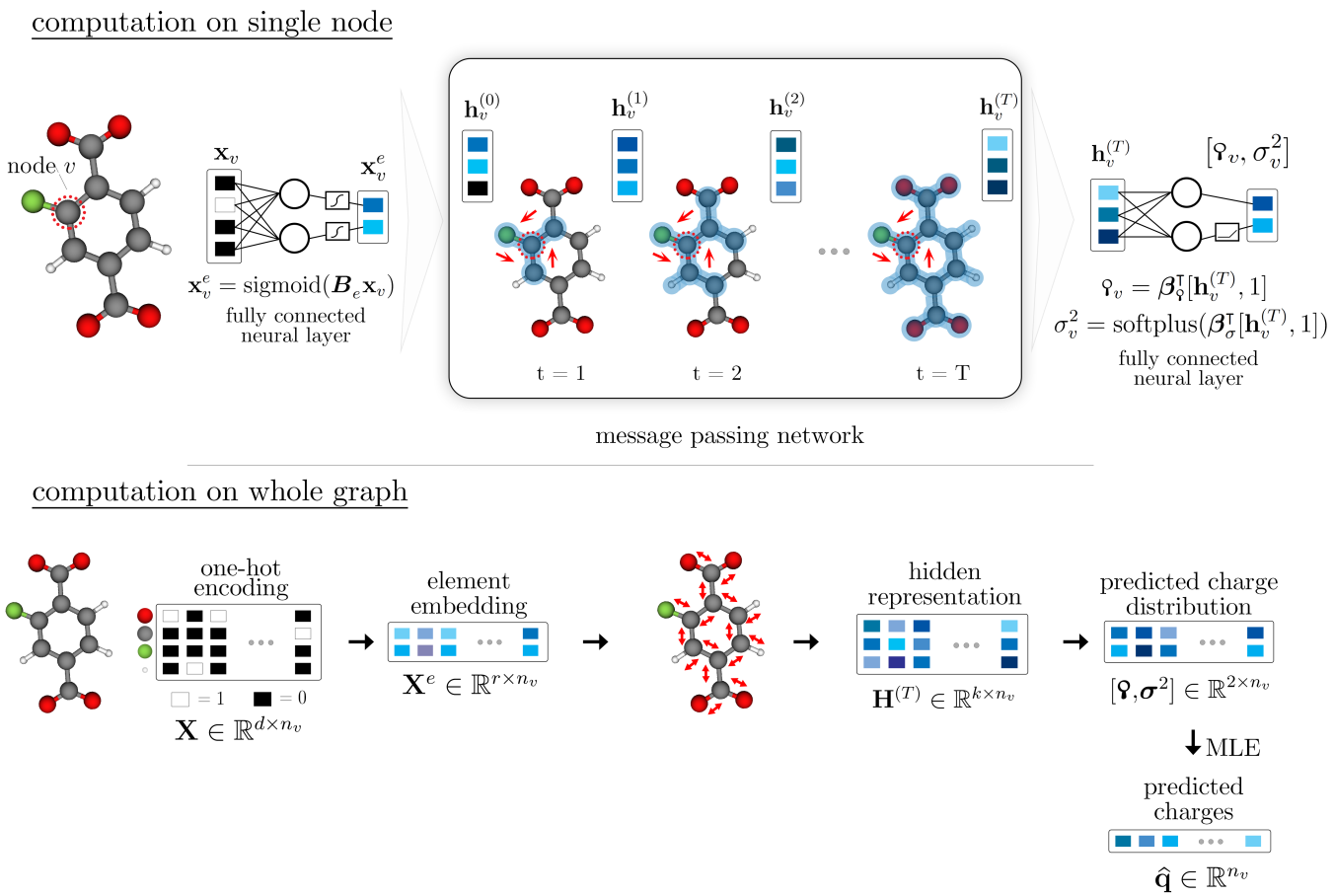


Figure 1: The architecture of our message passing neural network (MPNN) to predict partial charges on MOF atoms under a charge neutral constraint. (Top: node-level computations) First, the one-hot encoding of the atomic species of node v, $\mathbf{x}_v$, is mapped to an information-dense chemical element embedding $\mathbf{x}_v^e$ by a fully connected layer. The hidden node feature is initialized as $\mathbf{h}_v^{(0)}$ using $\mathbf{x}_v^e$. Next is the message passing phase, where, in each time step, every node shares information with its neighbors and updates its hidden representation. With more time steps, the hidden representation of the node, $\mathbf{h}_v^{(t)}$, captures a broader view of its local bonding environment. Modeling the probabilistic distribution of the charge on each node (within its local bonding environment) as Gaussian, a fully connected layer outputs the mean $\mathfrak{P}_v$ and variance σ_v^2 from the final, learned hidden representation $\mathbf{h}_v^{(T)}$. (Bottom: graph-level computations) The molecular graph G representing the MOF crystal structure is input to the MPNN. Each node is processed independently, as depicted on the top, culminating in the predicted means $\mathfrak{P}$ and variances $\boldsymbol{\sigma}^2$. Subsequently, a maximum likelihood estimation under the charge neutral constraint gives the predicted charges $\hat{\mathbf{q}}$.

its neighbors, summarizing the information received into a message $\mathbf{m}_v^{(t+1)}$. Specifically, the message received by node v is constructed as

$$\mathbf{m}_v^{(t+1)} = \boldsymbol{B}_m \sum_{u \in \mathcal{N}(v)} \mathbf{h}_u^{(t)} \tag{5}$$

where $\boldsymbol{B}_m$ is a learned matrix shared across all nodes and $\mathcal{N}(v)$ is the neighborhood of node v:

$$\mathcal{N}(v) = \{u \in \mathcal{V} | e_{uv} \in \mathcal{E}\}.$$

The hidden representation of node v is then updated by a GRU[67] (shared across all nodes) based on its message received and its current hidden representation:

$$\mathbf{h}_v^{(t+1)} = \text{GRU}(\mathbf{h}_v^{(t)}, \mathbf{m}_v^{(t+1)}). \tag{6}$$

See Sec. S4 for GRU details. The message passing phase is comprised of T such time steps.

At the end of the message passing phase, each node has a hidden representation $\mathbf{h}_v^{(T)}$ that encodes both its atomic species and its local bonding environment. Precisely, the *local bonding environment* of node v encoded in the hidden node feature $\mathbf{h}_v^{(T)}$ is the induced subgraph of G containing all nodes with geodesic distance less than or equal to T from node v. See Fig. 2.

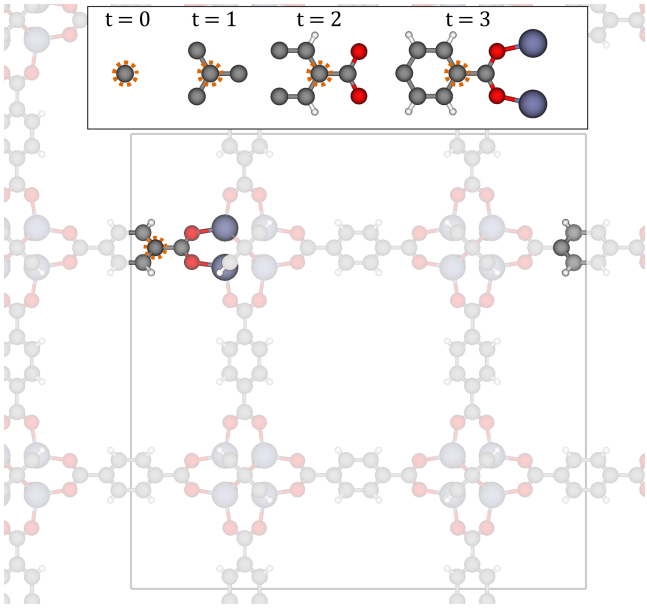


Figure 2: Illustration of the local bonding environment of node v after t message passing time steps (top), encoded in its hidden representation, $\mathbf{h}_v^{(t)}$. Focal node v, representing a carbon atom in IRMOF-1, is circled in orange. Edges across the unit cell boundary (gray square) are included to account for periodicity. As t increases, $\mathbf{h}_v^{(t)}$ contains a broader view of the bonding environment of the atom.

4.2.3 Charge prediction

Next, we use the learned representations of the local bonding environments of the nodes to predict their partial charges.

As opposed to directly predicting the charge on node v from $\mathbf{h}_v^{(T)}$, we instead view q_v as a random variable and model its conditional probability density as a Gaussian with mean $\wp_v = \wp_v(\mathbf{h}_v^{(T)})$ and variance $\sigma_v^2 = \sigma_v^2(\mathbf{h}_v^{(T)})$:

$$q_v|\mathbf{h}_v^{(T)} \sim N(\wp_v, \sigma_v^2). \tag{7}$$

i.e., we aim to learn and predict not only the typical charge of an atom within a given local bonding environment, but also its variance. The variance will be useful for adjusting the charges to enforce charge neutrality; charges of atoms with higher variance will be given more slack.

We use a fully connected layer comprised of two neurons, with weights $\boldsymbol{\beta}_\wp$ and $\boldsymbol{\beta}_\sigma$ shared across all nodes, to predict $\wp_v$ and σ_v^2, respectively, from the learned representation of the local bonding environment of node v, $\mathbf{h}_v^{(T)}$:

$$\wp_v = \boldsymbol{\beta}_\wp^\intercal[\mathbf{h}_v^{(T)}, 1] \tag{8}$$

$$\sigma_v^2 = \texttt{softplus}(\boldsymbol{\beta}_\sigma^\intercal[\mathbf{h}_v^{(T)}, 1]) \tag{9}$$

The softplus activation function ensures $\sigma^2 > 0$.

Finally, to arrive at the predicted charges $\hat{\mathbf{q}}$ on a given MOF, under the charge neutral constraint in eqn. 2, we invoke the maximum likelihood principle. Assuming that each q_v is conditionally independent given its local bonding environment and distributed according to eqn. 7, the log-likelihood $\mathcal{L}$ of observing charges $\hat{\mathbf{q}}$ on a given MOF is:

$$\mathcal{L}(\hat{\mathbf{q}}) = \sum_{v=1}^{n_v} \left(\log\left(\frac{1}{\sigma_v \sqrt{2\pi}} \right) - \frac{(\wp_v - \hat{q}_v)^2}{2\sigma_v^2} \right) \tag{10}$$

Maximizing $\mathcal{L}$ under the charge neutral constraint $\sum_{v=1}^{n_v} \hat{q}_v = 0$, we find (see S5):

$$\hat{q}_v = \wp_v - \frac{\sigma_v^2}{\sum_{u=1}^{n_v} \sigma_u^2} \sum_{u=1}^{n_v} \wp_u. \tag{11}$$

Interpreting the mean of the Gaussian in eqn. 7, $\wp_v$, as a before-constraint charge assignment for node v lends a useful interpretation of eqn. 11. To enforce charge neutrality, eqn. 11 adjusts the before-constraint charges by distributing the negative of the net before-constraint charge $\sum_{u=1}^{n_v} \wp_u$ to each atom in proportion to its variance σ_v^2. The idea is that, if the charge of an atom within its local bonding environment exhibits high variance among MOFs, then it

should be given more slack when adjusting it to achieve charge neutrality. If all local bonding environments exhibit the same variance, eqn. 11 reduces to uniformly distributing the before-constraint excess charge among the atoms. Eqn. 11, the last layer of our MPNN architecture, can be viewed as the *charge-correction layer*. By jointly learning $\boldsymbol{\beta}_{\text{ϙ}}$ and $\boldsymbol{\beta}_{\sigma}$, our network not only learns the typical charges of the atoms within their given local bonding environments, ϙ, but also the slack we should give them when adjusting them to enforce charge neutrality, through $\boldsymbol{\sigma}^2$.

Interestingly, there is a direct analogy between the variance σ_v^2 of the charge on a given atom in its given local bonding environment and the hardness (the second derivative of energy with respect to charge) used to enforce charge neutrality in Ref. 39 (compare eqn. 14 in Ref. 39 with eqn. 11); indeed, a soft atom will be given more slack for adjusting its charge to enforce charge neutrality.

4.3 Training of the MPNN

We define the loss function ℓ to train (i.e., identify the parameters of) our network as:

$$\ell = \frac{1}{N_v}\sum_{m=1}^{M}\ell_m, \qquad \ell_m = ||\hat{\mathbf{q}}_m - \mathbf{q}_m||_1, \qquad (12)$$

where N_v is the total number of nodes among all of the MOFs, M is the total number of MOFs, $\hat{\mathbf{q}}_m$ is the vector of charges on atoms of MOF m predicted by the MPNN by eqn. 11, $\mathbf{q}_m$ is the vector of (taken as ground-truth) DFT-derived, DDEC charges,[44] and $||\cdot||_1$ is the L1 norm. The loss ℓ in eqn. 12 is equivalent to the mean (over all nodes) absolute deviation (MAD) performance metric.

5 Results

Here, we train the MPNN in Fig. 1 and evaluate its performance. All computer codes (Python, Julia) to reproduce our work are available on Github at `github.com/SimonEnsemble/mpn_charges`.

5.1 The train, test, and validation datasets

Nazarian et al.[44] provide 2 932 MOF crystal structures with DFT-derived (PBE functional, DDEC method[32]) partial point charges assigned to each atom. We removed 607 duplicate MOFs (identified in Ref. 44) and nine erroneous MOFs (identified manually). Further, we automatically discarded MOF structures that, via our bonding algorithm, produced invalid bonding motifs (carbon atoms bonded to > 4 atoms, hydrogen atoms bonded to > 1 atom). See Sec. S3. Remaining are 2 266 charge-labeled MOFs. Fig. 3 shows the distribution of the partial charges, grouped by chemical element; many elements exhibit a high variance in charge, hinting that assigning charges to each atom solely based on its chemical element, without consideration of its bonding environment, will not give satisfactory charges. Fig. S1 shows the prevalence of chemical elements among the MOFs.

We randomly partitioned these 2 266 charge-labeled MOFs[44] into training, validation, and test sets (70/10/20%). The training set is used to directly tune the model parameters (weights and biases) by minimizing the loss in eqn. 12 over training examples via stochastic gradient descent. The validation set is used for hyper-parameter selection to avoid overfitting. The test set provides an unbiased evaluation of the performance of a final model whose parameters were fit using the training dataset. We note that chemical elements {Se, Hf, Cs, Pu, Ir} appear in only one MOF. Instead of discarding these MOFs containing these rare elements, we elected to place them in our training set, with the justification that we can learn about charges on other atoms from these MOFs.

5.2 Training and hyper-parameter tuning

We used the open source PyTorch[68] machine learning library to construct and train our MPNN.

To minimize the loss ℓ (see eqn. 12) during training, we use stochastic gradient descent (the

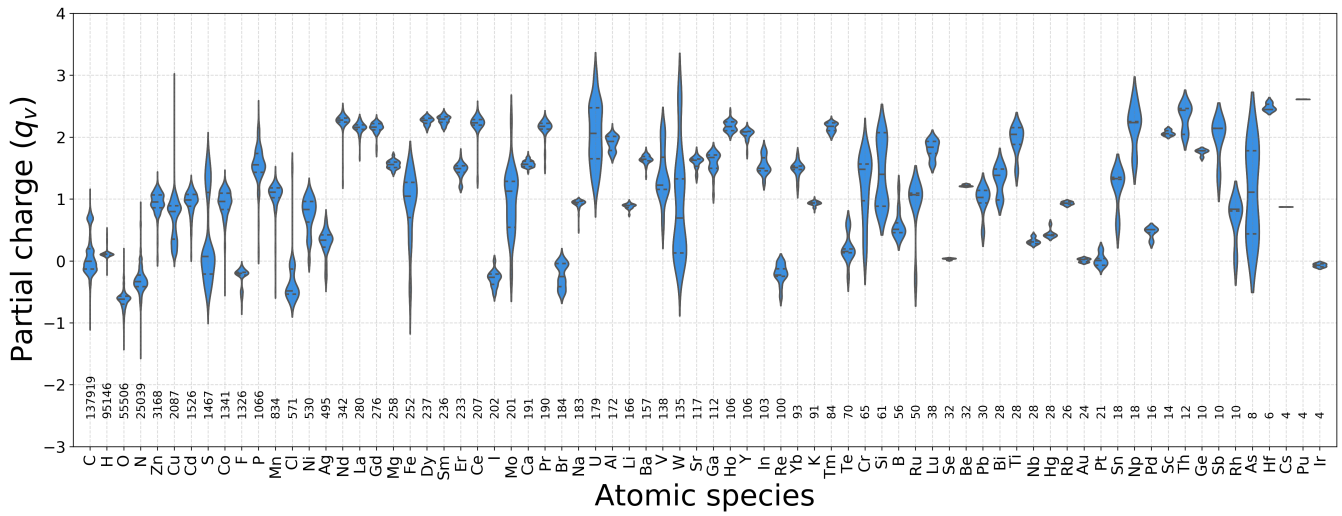


Figure 3: The distribution of partial charges, grouped by element, in the 2 266 charge-labeled MOFs[44] comprising our train, validation, and test set, visualized by violins. The number on the bottom is the total number of atoms in the dataset, grouped by element.

Adam optimizer with a learning rate of 0.005) and a batch size of 32 graphs. We use the validation set for early stopping to avoid overtraining. This is achieved by, while training, continuously monitoring the model performance on the validation set. Once no improvement is observed for 100 epochs, we stop the training and output the model with the best performance on the validation set. See Fig. S3 for an example learning curve.

The dimension of the element embeddings, r, dimension of the hidden node features, k, and number of message passing time steps, T, are hyper-parameters of our MPNN. We explored hyper-parameter space by changing one while holding the others fixed. See Sec S6. We found the performance of the MPNN to be largely insensitive to k and r for $k \geq 8$ and $r \geq 30$ (for fixed $r = 10$ and $k = 30$, respectively; see Fig. S2). On the other hand, we found the MPNN performance to be sensitive to T, which we discuss in Sec. 5.4. Based on our empirical hyper-parameter exploration, we select $T = 4$, $r = 10$, $k = 30$ since these hyperparameters led to the best performance on the validation dataset.

5.3 Performance

We evaluate the performance of our MPNN using the mean absolute deviation (MAD) over all nodes, equal to the loss in eqn. 12. For comparison, we consider the following benchmark models: (i) all charges are zero ($\hat{q}_v = 0$, $\forall v$), (ii) the charge of an atom is equal to the mean charge of atoms of that chemical element, with charge neutrality enforced by distributing excess charge among the atoms (a) uniformly and (b) proportional to its variance in the training set, as in eqn. 11, (iii) the I-QEq[53] charge equilibration method (MAD reported in Ref. 35). Tab. 2 summarizes the performance of our trained MPNN and these benchmark models. Results are the average of ten training/testing sessions with different (random) training, validation, and test splits. Our MPNN outperforms all baseline models, including the charge equilibration variant I-QEq,[53] which was the most consistent with DFT-derived, DDEC charges in the study by Ongari et al.[35] The MAD of our MPNN-assigned charges from the DFT-derived, DDEC (taken as ground truth) charges[44] is 0.025, a factor of four lower MAD than what I-QEq gives. Fig. 4 visualizes the joint distribution of the predicted charge $\hat{q}_v$ by our MPNN and the DFT-derived DDEC charge q_v; the density hugs the diagonal line.

Table 2: Performance benchmarks. The mean absolute deviation (MAD) on the test set for different charge assignment models/strategies.

Method (charge neutrality enforcement)	**MAD** mean (std)
$\hat{q}_v = 0, \forall\ v$	0.324 (7e−3)
element-mean (uniform dist'n excess charge)	0.154 (2e−3)
element-mean (variance-based dist'n excess charge)	0.153 (2e−3)
I-QEq[35,53§]	0.118[†]
MPNN (uniform dist'n excess charge)	0.026 (8e−4)
MPNN (variance-based dist'n excess charge)	0.025 (5e−4)

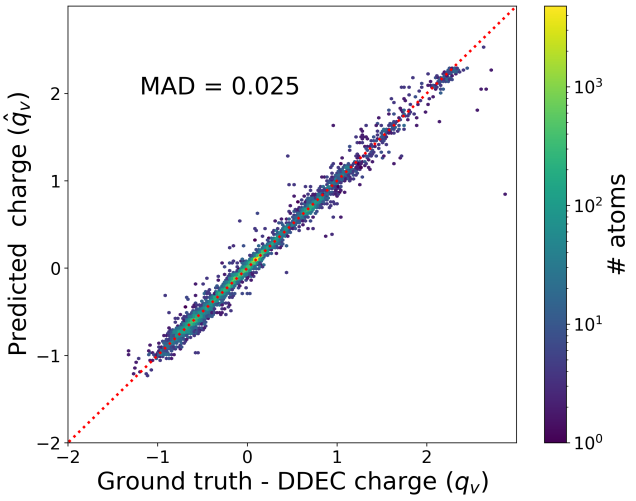


Figure 4: Parity plot showing a 2D histogram of the predicted charge $\hat{q}_v$ against the DFT-derived, DDEC charge q_v (treated as ground truth). Color depicts density of points. Diagonal line shows equality.

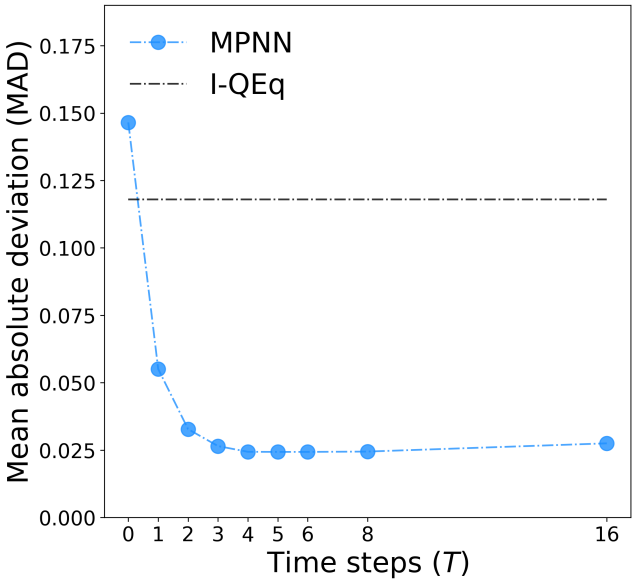


Figure 5: The performance of the MPNN (MAD on test set) as the number of message passing time steps, T, changes ($r = 10$, $k = 30$ fixed). As T increases, a broader view of the local bonding environment of each atom is used to predict its charge.

5.4 The effect of the number of message passing time steps, T

We investigate the effect of the number of message passing time steps, T, on the MPNN performance because, as Fig. 2 shows, T determines the scope of the local bonding environment of node v encoded in $\mathbf{h}_v^{(T)}$ and used to predict the charge on node v.

Fig. 5 shows the performance of our MPNN as T changes (with $r = 10$, $k = 30$ fixed). Without a message passing layer ($T = 0$), information is not passed between neighboring nodes, and the neural network learns to assign charge based only on the chemical element of the atom, irrespective of the atoms to which it is bonded. The MAD for $T = 0$ is 0.15, larger than the I-QEq method. Introducing a single message passing iteration ($T = 1$) to bring in information from immediate neighbors significantly improves the prediction ($T = 1$ MAD is 0.06). As we increase T, each node receives information from a longer geodesic distance; $\mathbf{h}_v^{(T)}$ encodes a broader scope of the bonding environment of the node; and, until $T = 4$, the predictions improve, albeit with diminishing returns. Increasing T beyond

4 slightly diminishes the performance, suggesting that the most useful information for charge prediction can be somewhat localized. Specifically, all nodes within a geodesic distance of $T = 4$ of a node appear sufficient for producing a quality prediction of the charge, albeit we fixed r and k for this analysis. Using overly large T can potentially lead to too much focus on the global structure of the MOF, diluting the useful local information. This supports our underlying hypothesis that the charge of any given atom in a MOF is largely dictated by its identity and local bonding environment.

5.5 Latent space of chemical elements

The chemical element embedding in eqn. 3 maps each chemical element into a low-dimensional, dense, information-rich representation of the chemical elements for initializing the hidden node features. To verify the MPNN has learned a meaningful element embedding, we visualize these $r = 10$-dimensional element embeddings $\mathbf{x}_v^e$ via Uniform Manifold Approximation and Projection (UMAP)[69] (hyper-parameters: number of neighbors: 8, minimum distance: 0.05). UMAP is a dimension reduction technique that aims to keep local and global structures exhibited by the data in the high-dimensional space intact in the low-dimensional representation. Fig. 6 visualizes the 2D embedding of each of the 74 chemical elements in the MOFs, colored by the average charge on that element in the DFT-derived, DDEC charge-assigned MOFs.[44] Judging from how nearby chemical elements tend to have a similar mean charge, it appears that the learned element embeddings indeed are encoding information predictive of partial charge. Interestingly, although the clustering according to the family in the periodic table to which the elements belong (see Fig. S5) is not as prominent as according to the mean partial charge, some clusters are recovered. For example, the alkali earth metals {Mg, Ca, Sr, Ba}, the alkali metals {Li, Na, K}, halogens {Br, Cl, I}, and many lanthanoids are clustered together, while the other periodic table families are more scattered.

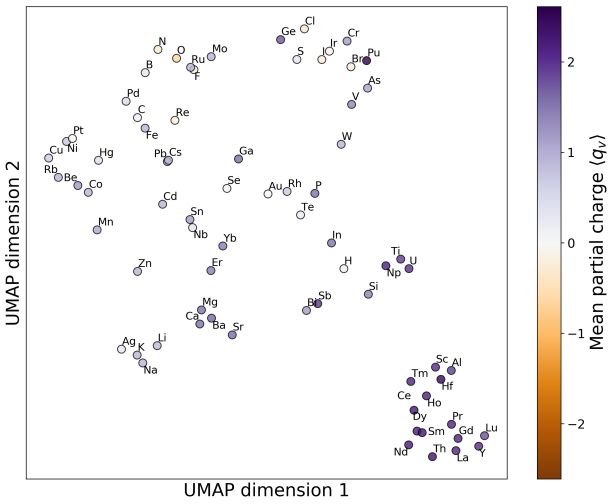


Figure 6: Visualization of the embeddings of the chemical elements learned by our MPNN and how they encode the mean charge of the element. We used UMAP[69] to reduce the dimension of the $r = 10$-dimensional embeddings $\mathbf{x}_v^e$ to the two dimensions shown here. Each point, representing the embedding of a chemical element, is colored according to the mean partial charge of that element in the training set. Note the mean partial charge tends to increase from top/top left to bottom right.

5.6 Enforcement of charge neutrality

We enforced charge neutrality on a given MOF through the design of our MPNN, by treating the charge on each atom within its local bonding environment as a conditionally independent random variable (eqn. 7), predicting the mean and variance of this distribution (eqns. 8 and 9), and using maximum likelihood to estimate the charges on the atoms of the MOF whilst satisfying charge neutrality (eqn. 11). A simpler strategy to enforce charge neutrality is to use a single (shared) neuron to directly predict a before-constraint charge on each atom from the learned node representation $\mathbf{h}_v^{(T)}$, then uniformly distribute the excess before-constraint charge on the MOF (see Sec. S7). The strategy to uniformly distribute excess before-constraint charge does not account for the tendency of

some atoms within certain local bonding environments to vary in their charge more than others; still, interestingly, this crude method of enforcing charge neutrality in the MPNN suffered in performance only marginally, with a MAD of 0.026.

Fig. S4 shows the distribution of the excess preliminary charge per atom, $\frac{1}{n_v}\sum_v \varrho_v$, among the MOFs for both strategies of enforcing charge neutrality. The standard deviation of $\frac{1}{n_v}\sum_v \varrho_v$ among the MOFs, with ϱ_v computed using eqn. 8, is only ≈ 0.02. This shows that relatively little charge neutrality correction of the before-constraint charges ϱ_v is needed (compare 0.02 with the MADs in Tab. 2). i.e., despite predicting charges based on the *local* bonding environment of the atoms, the MPNN outputs before-constraint charges ϱ_v that are "close" to satisfying the graph-level (global) charge neutrality constraint.

5.7 Deployment

For deployment, we re-trained our MPNN (with fixed hyper-parameters) using more training examples (2 040 MOFs plus 226 MOFs used for validation) to maximize its accuracy (setting aside a test set was necessary only for unbiased performance evaluation against different models). Note our MPNN (i) cannot predict charges for MOFs that contain chemical elements outside the set of 74 elements included in the DFT-derived, DDEC-charge assigned MOFs[44] we used for training and (ii) refrains from predicting charges on the elements {Se, Hf, Cs, Pu, Ir} since the training examples with these elements were too scarce to have confidence in predictions for these elements. See Tab. S6 for the list of viable chemical elements.

5.7.1 Public availability

Our deployment-ready MPNN model and our code to convert MOF crystal structures to graphs are available on Github (`github.com/SimonEnsemble/mpn_charges`) so the computational MOF community can assign high-quality charges to MOFs without performing computationally expensive electronic structure calculations. Moreover, our MPNN can easily handle MOFs with a large number of atoms, in contrast to periodic electronic structure calculations.

5.7.2 MPNN-charge-assigned CoRE MOFs

The updated computation-ready, experimental (CoRE) MOF dataset[7] contains ca. 14 000 structures, the majority of which are not present in the DDEC-charge-assigned set of Nazarian et al.[44] We used our deployment-ready MPNN to assign charges to each MOF in the v2 CoRE MOF database. To facilitate the use of these MPNN-assigned charges in molecular simulation studies, we provide `.cif` files on Github of the MPNN-charge-assigned v2 CoRE MOF structures.

As a caveat, the v2 CoRE MOF dataset is partitioned into two separate subsets based on the extent of solvent removal: (1) both bound and free solvent molecules removed and (2) only free solvent removed. The charge-assigned MOFs of Nazarian et al.[44] are based on structures in the v1 CoRE MOF dataset,[45] where both free and bound solvent molecules were removed. Consequently, the charge predictions by the MPNN may be less accurate on the subset of the v2 CoRE MOFs where only free solvent molecules were removed.

6 Discussion

We developed and trained a message passing neural network (MPNN)[50] to, in an end-to-end manner, learn representations of the local bonding environments of atoms within MOFs and, from these representations, predict the partial charges on the atoms of a MOF under a charge neutral constraint. The crystalline structure of the MOF, represented as an undirected graph with node features encoding the chemical elements, is directly input to the MPNN. The MPNN constructs features of the local bonding environments by sequentially passing information between bonded atoms. We trained and evaluated the performance of our MPNN by

leveraging 2 266 DFT-derived DDEC charge-labeled MOFs.[44] Our MPNN accurately predicts the partial charges on MOFs (mean absolute deviation from DDEC charges on test set, 0.025) while incurring orders of magnitude lower computational cost than performing electronic structure calculations and deriving charges from the electron density/electrostatic potential. We make our code and trained MPNN openly available to enable more accurate virtual screenings of thousands of MOFs, via molecular simulations using atomistic force fields,[8] for their adsorption-based applications in gas storage, separation/purification, and sensing. For convenience, we provide MPNN-charge-labeled v2 computation-ready, experimental MOF structures[7] in the widely used `.cif` format.

Notably, machine learning models can perform differently when employed on data drawn from a different distribution than the training data set. The training MOFs used for this MPNN are experimentally synthesized MOFs.[44,45] Consequently, we are confident that our MPNN network will perform well on experimentally synthesized MOFs. However, caution is warranted when using the MPNN on hypothetical MOFs sampled from a dramatically different distribution over MOF-space. For example, if hypothetical MOFs are constructed from elements in atomic environments that are rare in our training set of MOFs, then the accuracy of our MPNN could be reduced from what we report here.

Because we convert each MOF crystal structure to an undirected graph (with node features but not edge features), our MPNN will assign the same charges to (a) all conformations of the same MOF and (b) all interpenetrated isomers[70] of a MOF. To expand on (a), consider MOFs whose structures are flexible[71–73] and adopt different conformations depending on the temperature, imposed mechanical stress, and presence of adsorbed molecules.[73] To expand on (b), some MOFs form interpenetrated networks, and the level of interpenetration can be controlled.[70,74] Conceivably, the partial point charges that reproduce the electrostatic potential in the pores could differ depending on the conformation that the MOF adopts and its level of interpenetration. Our MPNN, however, would assign the same charges to the atoms of a MOF regardless of its conformation or level of interpenetration, since the graph representations of local bonding environments are invariant to flexing and interpenetration. To instead learn charges dependent on the conformation of and level of interpenetration in a MOF, we can encode the pairwise atomic distances between the atoms composing the MOF into edge features and employ MPNNs that handle edge features.[50] That is, we could represent each MOF as a fully connected graph and include two classes of edges: one for bonded atoms, and others for non-bonded atoms. Labeling edge e_{uv} with the distance between atom u and v would encode the 3D coordinates of the MOF into the graph.

Acknowledgement The authors acknowledge the National Science Foundation for support under grants No. 1920945 and No. 1521687.

Supporting Information Available

Supporting Info is available in a separate document. Our code, MPNN model, and MPNN-charge-assigned v2 CoRE MOFs are available at `github.com/SimonEnsemble/mpn_charges`.

References

(1) Furukawa, H.; Cordova, K. E.; O'Keeffe, M.; Yaghi, O. M. The chemistry and applications of metal-organic frameworks. *Science* **2013**, *341*, 1230444.

(2) Schoedel, A.; Ji, Z.; Yaghi, O. M. The role of metal–organic frameworks in a carbon-neutral energy cycle. *Nature Energy* **2016**, *1*, 1–13.

(3) Mueller, U.; Schubert, M.; Teich, F.; Puetter, H.; Schierle-Arndt, K.; Pastre, J. Metal–organic frameworks—prospective

industrial applications. *Journal of Materials Chemistry* **2006**, *16*, 626–636.

(4) Kreno, L. E.; Leong, K.; Farha, O. K.; Allendorf, M.; Van Duyne, R. P.; Hupp, J. T. Metal–organic framework materials as chemical sensors. *Chemical Reviews* **2012**, *112*, 1105–1125.

(5) Wang, C.; Liu, D.; Lin, W. Metal–organic frameworks as a tunable platform for designing functional molecular materials. *Journal of the American Chemical Society* **2013**, *135*, 13222–13234.

(6) Moghadam, P. Z.; Li, A.; Wiggin, S. B.; Tao, A.; Maloney, A. G.; Wood, P. A.; Ward, S. C.; Fairen-Jimenez, D. Development of a Cambridge Structural Database subset: a collection of metal–organic frameworks for past, present, and future. *Chemistry of Materials* **2017**, *29*, 2618–2625.

(7) Chung, Y. G.; Haldoupis, E.; Bucior, B. J.; Haranczyk, M.; Lee, S.; Zhang, H.; Vogiatzis, K. D.; Milisavljevic, M.; Ling, S.; Camp, J. S., et al. Advances, Updates, and Analytics for the Computation-Ready, Experimental Metal–Organic Framework Database: CoRE MOF 2019. *Journal of Chemical & Engineering Data* **2019**, *64*, 5985–5998.

(8) Sturluson, A.; Huynh, M. T.; Kaija, A. R.; Laird, C.; Yoon, S.; Hou, F.; Feng, Z.; Wilmer, C. E.; Colón, Y. J.; Chung, Y. G.; D, S.; C, S. The role of molecular modelling and simulation in the discovery and deployment of metal-organic frameworks for gas storage and separation. *Molecular Simulation* **2019**, *45*, 1082–1121.

(9) Boyd, P. G.; Lee, Y.; Smit, B. Computational development of the nanoporous materials genome. *Nature Reviews Materials* **2017**, *2*, 1–15.

(10) Cho, E. H.; Lyu, Q.; Lin, L.-C. Computational discovery of nanoporous materials for energy-and environment-related applications. *Molecular Simulation* **2019**, *45*, 1122–1147.

(11) Boyd, P. G.; Chidambaram, A.; García-Díez, E.; Ireland, C. P.; Daff, T. D.; Bounds, R.; Gładysiak, A.; Schouwink, P.; Moosavi, S. M.; Maroto-Valer, M. M., et al. Data-driven design of metal–organic frameworks for wet flue gas CO 2 capture. *Nature* **2019**, *576*, 253–256.

(12) Bucior, B. J.; Bobbitt, N. S.; Islamoglu, T.; Goswami, S.; Gopalan, A.; Yildirim, T.; Farha, O. K.; Bagheri, N.; Snurr, R. Q. Energy-based descriptors to rapidly predict hydrogen storage in metal–organic frameworks. *Molecular Systems Design & Engineering* **2019**, *4*, 162–174.

(13) Banerjee, D.; Simon, C. M.; Plonka, A. M.; Motkuri, R. K.; Liu, J.; Chen, X.; Smit, B.; Parise, J. B.; Haranczyk, M.; Thallapally, P. K. Metal–organic framework with optimally selective xenon adsorption and separation. *Nature Communications* **2016**, *7*, ncomms11831.

(14) Moghadam, P. Z.; Islamoglu, T.; Goswami, S.; Exley, J.; Fantham, M.; Kaminski, C. F.; Snurr, R. Q.; Farha, O. K.; Fairen-Jimenez, D. Computer-aided discovery of a metal–organic framework with superior oxygen uptake. *Nature Communications* **2018**, *9*, 1–8.

(15) Matito-Martos, I.; Moghadam, P. Z.; Li, A.; Colombo, V.; Navarro, J. A.; Calero, S.; Fairen-Jimenez, D. Discovery of an optimal porous crystalline material for the capture of chemical warfare agents. *Chemistry of Materials* **2018**, *30*, 4571–4579.

(16) Hulvey, Z.; Lawler, K. V.; Qiao, Z.; Zhou, J.; Fairen-Jimenez, D.; Snurr, R. Q.; Ushakov, S. V.; Navrotsky, A.; Brown, C. M.; Forster, P. M. Noble gas adsorption in copper trimesate,

HKUST-1: an experimental and computational study. *The Journal of Physical Chemistry C* **2013**, *117*, 20116–20126.

(17) Dubbeldam, D.; Frost, H.; Walton, K. S.; Snurr, R. Q. Molecular simulation of adsorption sites of light gases in the metal-organic framework IRMOF-1. *Fluid Phase Equilibria* **2007**, *261*, 152–161.

(18) Krause, S.; Bon, V.; Senkovska, I.; Stoeck, U.; Wallacher, D.; Többens, D. M.; Zander, S.; Pillai, R. S.; Maurin, G.; Coudert, F.-X., et al. A pressure-amplifying framework material with negative gas adsorption transitions. *Nature* **2016**, *532*, 348–352.

(19) Elsaidi, S. K.; Mohamed, M. H.; Simon, C. M.; Braun, E.; Pham, T.; Forrest, K. A.; Xu, W.; Banerjee, D.; Space, B.; Zaworotko, M. J., et al. Effect of ring rotation upon gas adsorption in SIFSIX-3-M (M= Fe, Ni) pillared square grid networks. *Chemical Science* **2017**, *8*, 2373–2380.

(20) Wilmer, C. E.; Farha, O. K.; Bae, Y.-S.; Hupp, J. T.; Snurr, R. Q. Structure–property relationships of porous materials for carbon dioxide separation and capture. *Energy & Environmental Science* **2012**, *5*, 9849–9856.

(21) Sikora, B. J.; Wilmer, C. E.; Greenfield, M. L.; Snurr, R. Q. Thermodynamic analysis of Xe/Kr selectivity in over 137000 hypothetical metal–organic frameworks. *Chemical Science* **2012**, *3*, 2217–2223.

(22) Sladekova, K.; Campbell, C.; Grant, C.; Fletcher, A. J.; Gomes, J. R.; Jorge, M. The effect of atomic point charges on adsorption isotherms of CO_2 and water in metal organic frameworks. *Adsorption* **2019**, 1–23.

(23) Zheng, C.; Liu, D.; Yang, Q.; Zhong, C.; Mi, J. Computational study on the influences of framework charges on CO_2 uptake in metal-organic frameworks. *Industrial & Engineering Chemistry Research* **2009**, *48*, 10479–10484.

(24) Walton, K. S.; Millward, A. R.; Dubbeldam, D.; Frost, H.; Low, J. J.; Yaghi, O. M.; Snurr, R. Q. Understanding inflections and steps in carbon dioxide adsorption isotherms in metal-organic frameworks. *Journal of the American Chemical Society* **2008**, *130*, 406–407.

(25) Hamad, S.; Balestra, S. R.; Bueno-Perez, R.; Calero, S.; Ruiz-Salvador, A. R. Atomic charges for modeling metal–organic frameworks: Why and how. *Journal of Solid State Chemistry* **2015**, *223*, 144–151.

(26) Potoff, J. J.; Siepmann, J. I. Vapor–liquid equilibria of mixtures containing alkanes, carbon dioxide, and nitrogen. *AIChE Journal* **2001**, *47*, 1676–1682.

(27) Dubbeldam, D.; Torres-Knoop, A.; Walton, K. S. On the inner workings of Monte Carlo codes. *Molecular Simulation* **2013**, *39*, 1253–1292.

(28) Watanabe, T.; Manz, T. A.; Sholl, D. S. Accurate treatment of electrostatics during molecular adsorption in nanoporous crystals without assigning point charges to framework atoms. *The Journal of Physical Chemistry C* **2011**, *115*, 4824–4836.

(29) Li, W.; Rao, Z.; Chung, Y. G.; Li, S. The Role of Partial Atomic Charge Assignment Methods on the Computational Screening of Metal-Organic Frameworks for CO_2 Capture under Humid Conditions. *ChemistrySelect* **2017**, *2*, 9458–9465.

(30) Altintas, C.; Keskin, S. Role of partial charge assignment methods in high-throughput screening of MOF adsorbents and membranes for CO_2/CH_4 separation. *Molecular Systems Design & Engineering* **2020**, *5*, 532–543.

(31) Campañá, C.; Mussard, B.; Woo, T. K. Electrostatic potential derived atomic

charges for periodic systems using a modified error functional. *Journal of Chemical Theory and Computation* **2009**, *5*, 2866–2878.

(32) Manz, T. A.; Sholl, D. S. Chemically meaningful atomic charges that reproduce the electrostatic potential in periodic and nonperiodic materials. *Journal of Chemical Theory and Computation* **2010**, *6*, 2455–2468.

(33) Argueta, E.; Shaji, J.; Gopalan, A.; Liao, P.; Snurr, R. Q.; Gómez-Gualdrón, D. A. Molecular Building Block-Based Electronic Charges for High-Throughput Screening of Metal–Organic Frameworks for Adsorption Applications. *Journal of Chemical Theory and Computation* **2018**, *14*, 365–376.

(34) Rappe, A. K.; Goddard III, W. A. Charge equilibration for molecular dynamics simulations. *The Journal of Physical Chemistry* **1991**, *95*, 3358–3363.

(35) Ongari, D.; Boyd, P. G.; Kadioglu, O.; Mace, A. K.; Keskin, S.; Smit, B. Evaluating charge equilibration methods to generate electrostatic fields in nanoporous materials. *Journal of Chemical Theory and Computation* **2018**, *15*, 382–401.

(36) Bleiziffer, P.; Schaller, K.; Riniker, S. Machine learning of partial charges derived from high-quality quantum-mechanical calculations. *Journal of Chemical Information and Modeling* **2018**, *58*, 579–590.

(37) Rai, B. K.; Bakken, G. A. Fast and accurate generation of ab initio quality atomic charges using nonparametric statistical regression. *Journal of Computational Chemistry* **2013**, *34*, 1661–1671.

(38) Martin, R.; Heider, D. ContraDRG: Automatic partial charge prediction by Machine Learning. *Frontiers in Genetics* **2019**, *10*, 990.

(39) Wang, Y.; Fass, J.; Stern, C. D.; Luo, K.; Chodera, J. Graph Nets for Partial Charge Prediction. *arXiv preprint arXiv:1909.07903* **2019**,

(40) Xu, Q.; Zhong, C. A general approach for estimating framework charges in metal-organic frameworks. *The Journal of Physical Chemistry C* **2010**, *114*, 5035–5042.

(41) Zheng, C.; Zhong, C. Estimation of framework charges in covalent organic frameworks using connectivity-based atom contribution method. *The Journal of Physical Chemistry C* **2010**, *114*, 9945–9951.

(42) Engler, M. S.; Caron, B.; Veen, L.; Geerke, D. P.; Mark, A. E.; Klau, G. W. Automated partial atomic charge assignment for drug-like molecules: a fast knapsack approach. *Algorithms for Molecular Biology* **2019**, *14*, 1.

(43) Zou, C.; Penley, D. R.; Cho, E. H.; Lin, L.-C. Efficient and Accurate Charge Assignments via Multi-Layer Connectivity-based Atom Contribution (m-CBAC) Approach. *The Journal of Physical Chemistry C* **2020**,

(44) Nazarian, D.; Camp, J. S.; Sholl, D. S. A comprehensive set of high-quality point charges for simulations of metal–organic frameworks. *Chemistry of Materials* **2016**, *28*, 785–793.

(45) Chung, Y. G.; Camp, J.; Haranczyk, M.; Sikora, B. J.; Bury, W.; Krungleviciute, V.; Yildirim, T.; Farha, O. K.; Sholl, D. S.; Snurr, R. Q. Computation-ready, experimental metal–organic frameworks: A tool to enable high-throughput screening of nanoporous crystals. *Chemistry of Materials* **2014**, *26*, 6185–6192.

(46) Wilmer, C. E.; Leaf, M.; Lee, C. Y.; Farha, O. K.; Hauser, B. G.; Hupp, J. T.; Snurr, R. Q. Large-scale screening of hypothetical metal–organic frameworks. *Nature Chemistry* **2012**, *4*, 83.

(47) Colón, Y. J.; Gómez-Gualdrón, D. A.; Snurr, R. Q. Topologically guided, automated construction of metal–organic

frameworks and their evaluation for energy-related applications. *Crystal Growth & Design* **2017**, *17*, 5801–5810.

(48) Boyd, P. G.; Woo, T. K. A generalized method for constructing hypothetical nanoporous materials of any net topology from graph theory. *CrystEngComm* **2016**, *18*, 3777–3792.

(49) Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Yu, P. S. A comprehensive survey on graph neural networks. *arXiv preprint arXiv:1901.00596* **2019**,

(50) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural message passing for quantum chemistry. Proceedings of the 34th International Conference on Machine Learning-Volume 70. 2017; pp 1263–1272.

(51) Bartók, A. P.; Kondor, R.; Csányi, G. On representing chemical environments. *Physical Review B* **2013**, *87*, 184115.

(52) Collins, C. R.; Gordon, G. J.; Von Lilienfeld, O. A.; Yaron, D. J. Constant size descriptors for accurate machine learning models of molecular properties. *The Journal of chemical physics* **2018**, *148*, 241718.

(53) Wells, B. A.; De Bruin-Dickason, C.; Chaffee, A. L. Charge equilibration based on atomic ionization in metal–organic frameworks. *The Journal of Physical Chemistry C* **2015**, *119*, 456–466.

(54) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M., et al. Analyzing learned molecular representations for property prediction. *Journal of Chemical Information and Modeling* **2019**, *59*, 3370–3388.

(55) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *Journal of Computer-aided Molecular Design* **2016**, *30*, 595–608.

(56) Stokes, J. M.; Yang, K.; Swanson, K.; Jin, W.; Cubillos-Ruiz, A.; Donghia, N. M.; MacNair, C. R.; French, S.; Carfrae, L. A.; Bloom-Ackerman, Z., et al. A Deep Learning Approach to Antibiotic Discovery. *Cell* **2020**, *180*, 688–702.

(57) Duvenaud, D. K.; Maclaurin, D.; Iparraguirre, J.; Bombarell, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. Advances in Neural Information Processing Systems. 2015; pp 2224–2232.

(58) Xie, T.; Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical Review Letters* **2018**, *120*, 145301.

(59) Coley, C. W.; Barzilay, R.; Green, W. H.; Jaakkola, T. S.; Jensen, K. F. Convolutional embedding of attributed molecular graphs for physical property prediction. *Journal of Chemical Information and Modeling* **2017**, *57*, 1757–1772.

(60) Jørgensen, P. B.; Jacobsen, K. W.; Schmidt, M. N. Neural message passing with edge updates for predicting properties of molecules and materials. *arXiv preprint arXiv:1806.03146* **2018**,

(61) Tang, B.; Kramer, S. T.; Fang, M.; Qiu, Y.; Wu, Z.; Xu, D. A self-attention based message passing neural network for predicting molecular lipophilicity and aqueous solubility. *Journal of Cheminformatics* **2020**, *12*, 1–9.

(62) Sanchez-Lengeling, B.; Wei, J. N.; Lee, B. K.; Gerkin, R. C.; Aspuru-Guzik, A.; Wiltschko, A. B. Machine Learning for Scent: Learning Generalizable Perceptual Representations of Small Molecules. *arXiv preprint arXiv:1910.10685* **2019**,

(63) Isayev, O.; Oses, C.; Toher, C.; Gossett, E.; Curtarolo, S.; Tropsha, A. Universal fragment descriptors for predicting properties of inorganic crystals. *Nature Communications* **2017**, *8*, 1–12.

(64) Cordero, B.; Gómez, V.; Platero-Prats, A. E.; Revés, M.; Echeverría, J.; Cremades, E.; Barragán, F.; Alvarez, S. Covalent radii revisited. *Dalton Transactions* **2008**, 2832–2838.

(65) Virtanen, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **2020**, *17*, 261–272, DOI: doi.org/10.1038/s41592-019-0686-2.

(66) Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493* **2015**,

(67) Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259* **2014**,

(68) Paszke, A. et al. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc., 2019; pp 8024–8035.

(69) McInnes, L.; Healy, J.; Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* **2018**,

(70) Jiang, H.-L.; Makal, T. A.; Zhou, H.-C. Interpenetration control in metal–organic frameworks for functional applications. *Coordination Chemistry Reviews* **2013**, *257*, 2232–2249.

(71) Coudert, F.-X. Responsive metal–organic frameworks and framework materials: under pressure, taking the heat, in the spotlight, with friends. *Chemistry of Materials* **2015**, *27*, 1905–1916.

(72) Ortiz, A. U.; Boutin, A.; Fuchs, A. H.; Coudert, F.-X. Metal–organic frameworks with wine-rack motif: What determines their flexibility and elastic properties? *The Journal of Chemical Physics* **2013**, *138*, 174703.

(73) Boutin, A.; Springuel-Huet, M.-A.; Nossov, A.; Gedeon, A.; Loiseau, T.; Volkringer, C.; Férey, G.; Coudert, F.-X.; Fuchs, A. H. Breathing Transitions in MIL-53 (Al) Metal–Organic Framework Upon Xenon Adsorption. *Angewandte Chemie International Edition* **2009**, *48*, 8314–8317.

(74) Eddaoudi, M.; Kim, J.; Rosi, N.; Vodak, D.; Wachter, J.; O'Keeffe, M.; Yaghi, O. M. Systematic design of pore size and functionality in isoreticular MOFs and their application in methane storage. *Science* **2002**, *295*, 469–472.