Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2025, **4**, 2972

Beyond training data: how elemental features enhance ML-based formation energy predictions

Hamed Mahdavi, *[a] Vasant Honavar[a] and Dane Morgan[b]

Quantum mechanics (QM) based modeling allows for accurate prediction of molecular and atomic interactions, enabling simulations of many materials and chemical properties. However, the high computational cost of QM models leads to a need for faster computational methods to study atomic-scale interactions. Graph Neural Networks fit to QM calculations have been used as a computationally efficient alternative to QM. Still, generalization to diverse unseen compounds is challenging due to the many possible chemistries and structures. In this work, we demonstrate the effectiveness of utilizing element features in facilitating generalization to compounds containing completely new elements in the dataset. Our findings show that we can even randomly exclude up to ten percent of the elements from the dataset without significantly compromising the model's performance.

Received 5th May 2025
Accepted 16th August 2025

DOI: 10.1039/d5dd00182j

rsc.li/digitaldiscovery

Introduction

Quantum mechanics (QM) provides a highly accurate description of atomic-scale interactions, but the computational complexity of quantum mechanical methods hampers the ability to thoroughly investigate all possible arrangements of different chemical elements.[1] There is therefore a need for more computationally efficient approaches to studying molecular interactions. In response to this challenge, researchers have increasingly leveraged tools from machine learning, particularly Graph Neural Networks (GNNs), as an alternative to QM.[2] However, the combinatorial nature of chemical interactions makes it difficult to gather a comprehensive training dataset of different chemical species and environments, leading to a requirement for predictive models that can extrapolate beyond the training data. Ensuring these models remain reliable when encountering chemistries or configurations not seen during training is a core challenge.[3] In this context, out-of-distribution (OoD) generalization is potentially very useful as it involves developing models that can make accurate predictions for elements or interactions that were not present during the training phase. A promising approach to the OoD issue is to incorporate physical knowledge and concepts, such as symmetries, into machine learning models to improve generalization.[4–13] Another relevant body of work studies the generalization benefits of learned and QM descriptors in machine learning based prediction of material properties.[14,15] Li *et al.*[14] provide a simple and useful flowchart that helps to decide when using QM descriptors in GNN models could be helpful. Chen *et al.*[15] demonstrate the effectiveness of pre-trained ML-based atomic descriptors for formation enthalpy prediction. Their model, trained on molecules containing up to 11 heavy atoms, is capable of predicting the formation enthalpy of test molecules with as many as 42 heavy atoms, achieving a low mean absolute error. The other line of work concentrates on data-centric methods: Smith *et al.*[16] introduced active learning protocols (*e.g.*, in ANI-1x) to curate diverse training sets, significantly improving coverage of chemical space. Uncertainty quantification techniques (ensembles, Bayesian NNs) are also frequently employed to detect when a model is querying an unseen region, enabling on-the-fly error mitigation or data augmentation.[17–20] In the present work, we consider how the physical knowledge embedded in elemental descriptors can help with OoD predictions in machine learning based potentials. This is particularly beneficial in scenarios where one seeks to develop universal interatomic potentials applicable across the entire periodic table.[21–24] The majority of machine learning models for molecular tasks represent each element either with a one-hot encoding or a real vector.[4–8,10–13,25] Li *et al.*[26] conduct a comprehensive evaluation of multiple machine learning approaches across 700 OoD tasks. Their results indicate that common machine learning models applied to materials datasets demonstrate strong generalization capabilities, successfully predicting targets even for tasks that involve new chemical elements or structural groups absent in the training set. By examining the representation spaces of materials, the authors find that test samples from tasks showing high predictive accuracy generally reside within the regions covered by the training data. Conversely, samples from tasks with lower performance levels tend to lie outside the training domain. Additionally, their analysis reveals that the generalization

[a]*Department of Computer Science and Engineering, Pennsylvania State University, State College, PA 16801, USA. E-mail: hmm5834@psu.edu; vuh14@psu.edu*
[b]*Department of Material Science and Engineering, University of Wisconsin Madison, Wisconsin, WI 53706, USA. E-mail: ddmorgan@wisc.edu*

behavior on OOD tasks does not strictly follow traditional scaling laws. Indeed, enlarging the training dataset or prolonging the training period leads to minimal improvements and may even worsen the performance on particularly challenging OOD prediction tasks. While the models in Li *et al.*[26] exhibit a degree of generalization to compounds with OoD elements, a significant performance disparity persists between models trained with and without exposure to specific elements or elemental sets. Elements possess strongly correlated characteristics that can be represented by a set of features that capture their chemical relationships. For instance, the periodic table illustrates established patterns that correlate the properties of elements with one another. Prior studies have attempted to integrate element features to improve material tasks prediction accuracy in i.i.d (independent identically distributed) setting where the test data distribution is the same the training data.[27,28] This study provides novel insights into the factors contributing to diminished model performance on compounds containing OoD elements. Furthermore, we demonstrate incorporating elemental features enhances predictive capabilities in OoD scenarios. Specifically, this approach significantly improves the prediction of formation energies for compounds with novel unseen elements during, often yielding performance comparable to that observed when the element is abundant in the training data. Additionally, we demonstrate that elemental features also improve generalization behavior and scaling laws for OoD elements.

Dataset and models description

Dataset

We used the mp_e_form dataset from Matbench v0.1 test-suite.[29] This dataset includes 132752 compound structures and their associated formation energies, all selected from the Materials Project data.[30] All compounds in this dataset are inorganic and the formation energies are calculated using density functional theory (DFT) at the Generalized Gradient Approximation (GGA) level, which we will call the DFT-GGA method.[31] We used the Python package XenonPy to gather a 94 $\times$ 58 feature matrix for the first 94 elements in the periodic table. The feature list includes many property values collected as part of developing the Xenopy package, including atomic radius, van der Waals radius, atomic number, and period. Table 1 presents the complete list of features we utilized in this project. We denote this feature matrix with **H** during the paper.

Models

As a baseline and use case for our approach, we use the SchNet[7] and MACE[32] models. SchNet is an invariant molecular energy prediction and atomic force modeling framework. Its architecture ensures that the predicted energy is invariant to the molecule's orientation and atom indexing, while the predicted forces are equivariant, meaning they rotate consistently with the molecular structure.

The core of SchNet involves continuous-filter convolution (cfconv) layers, which apply Gaussian radial basis functions and multi-layer perceptrons to atomic distances to obtain the hidden representation of each atom. Using only the atomic distances ensures the model's invariance to rotations and translations of coordinates.

MACE uses equivariant message passing, making the model more powerful than invariant ones. The main idea of MACE lies in decomposing many-body messages between atoms into a novel linear expansion of higher-order features, allowing efficient calculation of these messages. Similar to SchNet, which

Table 1 Complete list of the 58 element-level features used in this project. The set includes basic identifiers (atomic number, period, Mendeleev number), various size scales (covalent, metallic, and van der Waals radii), thermodynamic quantities (enthalpies, heat capacities), mechanical moduli, and selected DFT-derived ground-state properties

Selected elemental features		
Atomic number	Atomic radius	Atomic radius[33]
Atomic volume	Atomic weight	Boiling temperature
Bulk modulus	C_6 dispersion coefficient (Gould–Bučko)	Covalent radius[34]
Covalent radius (single bond)[35]	Covalent radius (double bond)[36]	Covalent radius (triple bond)[37]
Covalent radius (slater)	Density at 295 K	Dipole polarizability
Generic electronegativity	Electron affinity	Allen electronegativity scale
Ghosh electronegativity scale	Pauling electronegativity scale	First ionisation energy
Fusion enthalpy	DFT band–gap energy (0 K)	DFT energy per atom (0 K)
Estimated BCC lattice constant (DFT)	Estimated FCC lattice constant (DFT)	DFT magnetic moment (0 K)
DFT volume per atom (0 K)	HHI production index	HHI reserves index
Mass specific heat capacity	Molar specific heat capacity	Atom volume in ICSD database
Evaporation heat	Heat of formation	Unit-cell lattice constant
Mendeleev number	Melting point	Molar volume
Total unfilled electrons	Total valence electrons	Unfilled d electrons
Valence d electrons	Unfilled f electrons	Valence f electrons
Unfilled p electrons	Valence p electrons	Unfilled s electrons
Valence s electrons	Period in the periodic table	Specific heat at 20 °C
Thermal conductivity at 25 °C	van der Waals radius	van der Waals radius (Alvarez)
van der Waals radius (MM3)	van der Waals radius (UFF)	Speed of sound
Polarizability (instantaneous dipoles)		

uses an embedding matrix to represent each element, MACE can also process any vector-shaped representation of elements. To ensure consistency, we also used vector-shaped embeddings for MACE, where these embeddings are optimized during training. A detailed description of the model hyper-parameters we used for training is included in the Appendix section.

Problem definition and setting

Our research focuses on investigating how a machine learning model performs when it encounters compounds containing previously unseen elements, which were not part of the original training dataset. We are specifically focused on predicting the formation energies of compounds from a large training dataset of structures and their energies. We have defined an OoD scenario in which we remove all compounds containing a specific set of elements, such as *cobalt*, from the training set. The goal is to predict the formation energies of structures containing these excluded elements.

A recent comprehensive study by Li *et al.*[26] systematically evaluated the OoD generalization capabilities of various machine learning models, from tree ensembles to graph neural networks and large language models, across over 700 tasks involving unseen chemistry or crystal structures in materials science. Their findings indicate that many models, including simpler ones, surprisingly exhibit robust generalization across most heuristically defined OoD tasks, such as leaving out a single element or entire groups/periods. They attribute this, in part, to the fact that many representations implicitly capture elemental relationships, allowing for effective interpolation even when specific elements are absent from training. However, Li *et al.*[26] also identified specific elements (*e.g.*, H, F, O) and tasks that remain genuinely challenging, where test data lie significantly outside the domain covered by the training data in the representation space (representationally OOD). Crucially, they found that for these truly difficult OoD tasks, standard scaling approaches, like increasing training data size (of seen elements) or training time, provide marginal benefits or can even degrade performance, highlighting the limitations of standard training paradigms for extrapolation.

As highlighted Li *et al.*,[26] while machine learning models can exhibit notable generalization even when encountering some previously unseen elements, a substantial performance gap often persists, particularly for elements that are chemically or structurally distinct from the training data. Addressing this remaining challenge in OoD generalization is critical for developing truly predictive and transferable materials models. This work therefore investigates the underlying reasons for this performance discrepancy, focusing specifically on the limitations inherent in standard graph neural network frameworks that utilize conventional learnable element embeddings ($\mathbf{E}$). Furthermore, we propose and evaluate a method designed to significantly mitigate this gap and enhance predictive accuracy for compounds containing such OoD elements. To establish the specific technical context for this study, we consider having access to a dataset of compounds and their formation energies. The i-th compound in this dataset, $\mathscr{D}$, is represented by the triplet $(\mathbf{z}_i,\mathbf{x}_i,y_i)_{i=1}^{N}$, where $\mathbf{z}_i$ represents the set of atom types in the compound, $\mathbf{x}_i$ denotes the corresponding 3-dimensional coordinates of each atom, and y_i is the formation energy. For the purpose of potential energy estimation, a typical graph neural network model, f_θ, takes as inputs the atomic types $\mathbf{z}_i$, the atomic coordinates $\mathbf{x}_i$, and an $m \times d$ learnable embedding matrix $\mathbf{E}$ used to represent the m unique elements with d-dimensional vectors. The model $f_\theta(\mathbf{z}_i, \mathbf{x}_i, \mathbf{E})$ then outputs an estimated potential energy $\hat{y}$. Conventionally, the network parameters θ and the element embeddings matrix $\mathbf{E}$ are optimized jointly during the training process.

The problem we are considering is how models, represented by $f_\theta(\mathbf{z}_i, \mathbf{x}_i, \mathbf{E})$, can be best developed to effectively generalize to data containing new elements. In a standard implementation, achieving this type of generalization reliably is challenging. As depicted in Fig. 1, this limitation arises because if an element, denoted as e, is absent from the training set, its corresponding row in the embedding matrix $\mathbf{E}$ is not updated, as it receives no gradient during the backward pass. This lack of updating means the embedding vector retains no learned information specific to element e, leading to inferior predictions for compounds containing this unseen element compared to predictions for elements present in the training data. In this study, we investigate the embedding matrix $\mathbf{E}$ and demonstrate the benefits of utilizing pre-defined element features within the embeddings, in contrast to relying solely on a randomly initialized embedding matrix.

To demonstrate the benefits of integrating element features for OoD generalization, we present a scenario where none of the compounds in the training set include any elements from the predefined set of elements $\mathscr{E}$, in their chemical formulas. Conversely, all compounds in the test set include at least one of the elements from $\mathscr{E}$. The objective is to effectively generalize to the test set, despite the lack of samples containing elements from $\mathscr{E}$ in their formulas during training.

To create the specific datasets for this task, we assume access to the dataset $\mathscr{D}$, which has been randomly divided into the training set $\mathscr{D}^{\text{Train}}$ and the test set $\mathscr{D}^{\text{Test}}$ using an i.i.d. random split, where $|\mathscr{D}| = |\mathscr{D}^{\text{Train}}| + |\mathscr{D}^{\text{Test}}|$. To prepare the training set for the OoD task, we randomly select a set of elements $\mathscr{E}$ and exclude

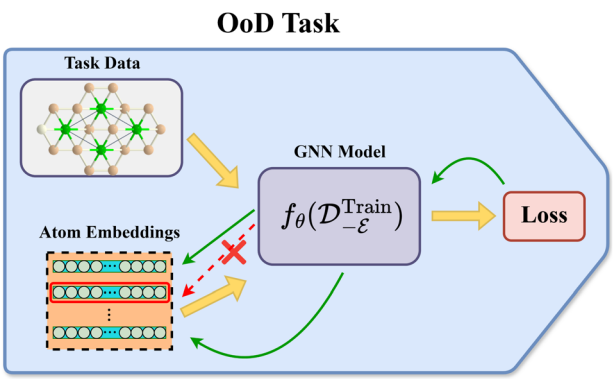


Fig. 1 Since OoD elements are absent from the training dataset, their corresponding embedding vectors receive no backpropagated gradients and thus remain static throughout the optimization process.

all compounds containing members of $\mathscr{E}$ from $\mathscr{D}^{\text{Train}}$, denoting it as $\mathscr{D}^{\text{Train}}_{-\mathscr{E}}$. Conversely, for the test set, we retain only compounds that include at least one member of $\mathscr{E}$ in their chemical formula and exclude all other compounds, denoted as $\mathscr{D}^{\text{Test}}_{\mathscr{E}}$.

Diagnosis and the proposed solution

To overcome the problem of OoD generalization to new elements defined in the previous section, we take a closer look at the embedding matrix **E**. Before we can approach a solution, however, we need to answer a few key questions related to how embeddings work in the context of GNNs for material compound energies. In particular, we seek to answer:

(i) How does a model perform in an OoD scenario if it uses the embedding matrix directly without prior knowledge of the elements?

(ii) Do the learned embedding vectors contain chemistry?

(iii) And finally, is there a fix to help the model generalize to unseen elements after it has been trained?

How does a model perform in an OoD scenario if it uses the embedding matrix directly without prior knowledge of the elements?

The goal of this section is to quantify the performance degradation observed when standard GNN element embedding approaches encounter OoD elements absent from the training data. While recent comprehensive studies, such as Li *et al.*,[26] have shown that models can often exhibit a degree of generalization to compounds containing unseen elements (likely due to learned chemical similarities), a significant performance gap typically persists compared to models trained with access to those elements. To illustrate and quantify this performance difference in our specific setup, we examine a leave-one-element-out OoD task. We select Germanium (Ge) as a representative case, although similar performance characteristics are expected for many other elements excluded from training. Specifically, we remove all compounds containing Ge from the training set to obtain $\mathscr{D}^{\text{Train}}_{-\{\text{Ge}\}}$ and train the MACE and SchNet models on this reduced dataset. We then evaluate these models on the test set containing Ge compounds, $\mathscr{D}^{\text{Test}}_{\{\text{Ge}\}}$. This performance is critically compared against that of the same models trained on the complete dataset, $\mathscr{D}^{\text{Train}}$, evaluated on the identical test data $\mathscr{D}^{\text{Test}}_{\{\text{Ge}\}}$. As demonstrated in Fig. 2, excluding Ge from the training set results in substantially higher Mean Absolute Error (MAE) and lower coefficient of determination (R^2) values when predicting properties for Ge-containing compounds during test time, indicating a significant degradation in performance for both MACE and SchNet models. This highlights the significant performance penalty incurred when relying on standard embeddings for elements unseen during training, even if some minimal generalization might be present, motivating the need for improved OoD strategies.

Do the learned embedding vectors contain chemistry?

This section investigates whether, under standard randomly initialized embeddings, chemical information is being integrated into the embeddings during training. In particular, we seek to check if training process is encoding information similar to that in our elemental property vectors. Additionally, we examine whether the relative positions of the element embedding vectors reflect any meaningful chemistry. We explore these questions as they might be useful for designing an approach that will generalize to new elements. If the embeddings lack chemical information, the only solution in scenarios with new elements in the test set is to retrain the model, and we might not always have enough data for the new elements to retrain our model. For the experiments of this section, we assume access to an embedding matrix **E**, which is trained on materials project data using mean absolute loss along with a model. For the models, we use SchNet[7] and MACE[32] (as described in Dataset and Models Description section).

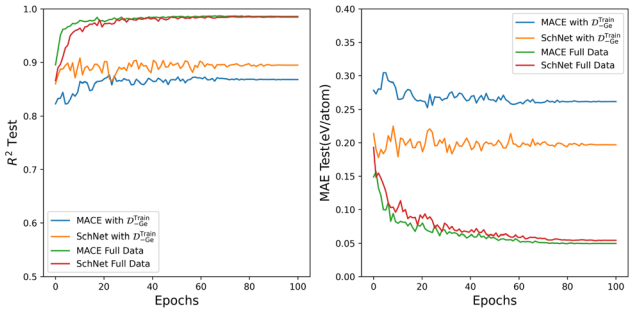


Fig. 2 Quantifying the performance gap for an unseen element (Ge) using standard embeddings. The plot compares test metrics on Ge-containing compounds ($\mathscr{D}^{\text{Test}}_{\{\text{Ge}\}}$) achieved by models trained with the full dataset *versus* models trained after removing all Ge-containing compounds ($\mathscr{D}^{\text{Train}}_{-\{\text{Ge}\}}$). A marked increase in Mean Absolute Error (MAE) and a corresponding decrease in the coefficient of determination (R^2) are observed when Ge is excluded from training, demonstrating the performance deficit when generalizing to unseen elements.

Assessing E–H similarity as a proxy for chemistry

One way to investigate the embeddings of elements is by examining the relative similarity of different elements. One might expect that chemically similar elements should have similar embeddings, analogous to word embeddings in natural language models.[38] We know that chemically similar elements have similar feature vectors in **H**. By examining the most similar elements to each element using **H** with a similarity measure such as cosine distance or negative l_2 distance, we can identify chemically similar elements. We can also list the set of similar elements using the same logic for the embedding matrices that are trained with MACE and SchNet. Table 2 represents the top-4 similar elements using negative l_2 distance for 8 elements using the embedding matrix **E** trained with MACE, the embedding matrix **E** trained with SchNet, and finally **H**. We can see the expected chemical correlations in Table 2c, *e.g.*, where elements similar to Cu include Ag, another noble metal, elements similar to Fe include other transition metals, elements similar to Ca include multiple other column II elements, and elements similar to P include other column VA and anion species.

However, such chemical correlation is not very apparent for Tables 2a and b. Looking at the Table 2a, Cu has Ag close to it,

Table 2 We have listed top-4 similar elements for 8 different elements using the negative l_2 distance for feature matrix H and embedding matrix E. As we can see, top-4 similar elements based on H in (c) are chemically meaningful, while the chemistry in (a) is less visible, and there is no pattern in the top-4 similar elements for (b)

Element	Top-4 similar elements			
(a) E for MACE				
Cu: 29	H: 1	Ag: 47	Ru: 44	Co: 27
Ti: 22	H: 1	Zr: 40	Ta: 73	Si: 14
Fe: 26	H: 1	Co: 27	Be: 4	O: 8
Ca: 20	Sr: 38	H: 1	Ba: 56	Y: 39
Si: 14	H: 1	S: 16	P: 15	Al: 13
Pb: 82	H: 1	Rb: 37	As: 33	O: 8
P: 15	Si: 14	H: 1	S: 16	O: 8
(b) E for SchNet				
Cu: 29	Na: 11	U: 92	S: 16	Br: 35
Ti: 22	U: 92	Cs: 55	Br: 35	Sc: 21
Fe: 26	Re: 75	K: 19	Sc: 21	Br: 35
Ca: 20	K: 19	N: 7	Na: 11	Sc: 21
Si: 14	Na: 11	Sr: 38	N: 7	O: 8
Pb: 82	Cs: 55	Ra: 88	Ba: 56	U: 92
P: 15	Au: 79	U: 92	Sc: 21	Y: 39
(c) H				
Cu: 29	Ag: 47	Cr: 24	Rh: 45	Ni: 28
Ti: 22	V: 23	Zr: 40	Sc: 21	Mn: 25
Fe: 26	Co: 27	Ni: 28	V: 23	Ti: 22
Ca: 20	Sr: 38	Mg: 12	Ba: 56	Na: 11
Si: 14	Al: 13	P: 15	Ge: 32	Ga: 31
Pb: 82	Bi: 83	Po: 84	Tl: 81	At: 85
P: 15	S: 16	Cl: 17	O: 8	N: 7

which is quite reasonably chemically, but the connection to Ru is not clear. Co shares some similarity with Cu, but H is completely irrelevant in that row. Fe has H and O in the same row, which are entirely unrelated to Fe. Be is also quite distinct, and the only obviously chemically relevant element is Co. The Table 2a sometimes contains similar elements in each row, but it reflects much less chemistry compared to the Table 2c. It also has some strange features, *e.g.*, H is included in all rows without any obvious reason. The rows in Table 2b appear random. For example, Cu has neighbor elements from many other columns in the periodic table with widely varying chemical nature, and no other noble metals. A comparable lack of chemical similarity can be seen for the nearest neighbors of other elements. To determine if something is missed by this simple analysis of a few cases we check if **E** encodes chemistry in ways similar to **H** with a more systematic approach. Specifically, we aim to answer the following question: If a pair of elements are similar with respect to representations from **H** using a fixed similarity metric, are they also similar with respect to features from **E**?

To estimate this quantitatively, we calculate the top-k similar elements for each element using **H** and **E**. If **E** encodes chemistry like **H**, then the two sets of top-k similar elements should be similar and have significant overlap. Intuitively, the amount of overlap between two sets shows how similar are the closest neighbours of a fixed element with respect to **E** and **H**. Formally, we used the following procedure to assess the overlap of similarity determined by **H** and **E**:

1. Since the rows of **E** and **H** represent elements, we calculated the similarity between each pair of elements using a similarity measure such as cosine similarity or negative l_2 distance, using **H** and **E**. In other words, for each pair of elements like i, j, we calculate $\mathrm{Sim}(\mathbf{E}_i, \mathbf{E}_j)$ and $\mathrm{Sim}(\mathbf{H}_i, \mathbf{H}_j)$ and, where $\mathrm{Sim}(\cdot, \cdot)$ is negative l_2 or cosine distance.

2. We save the obtained similarities in $m \times m$ matrices $S^{\mathbf{E}}$ and $S^{\mathbf{H}}$ respectively. So we have:

$$S_{ij}^{\mathbf{E}} = \mathrm{Sim}(\mathbf{E}_i, \mathbf{E}_j),\ S_{ij}^{\mathbf{H}} = \mathrm{Sim}(\mathbf{H}_i, \mathbf{H}_j) \tag{1}$$

3. Using the obtained pairwise similarities, we calculate the top-k similar elements for each element using the numbers from $S^{\mathbf{E}}$ and $S^{\mathbf{H}}$. We denote the set of top-k similar elements to element e using matrix **H** with $\mathrm{Top}_k^{\mathbf{H}}(e)$. We define $\mathrm{Top}_k^{\mathbf{E}}(e)$ in a similar manner.

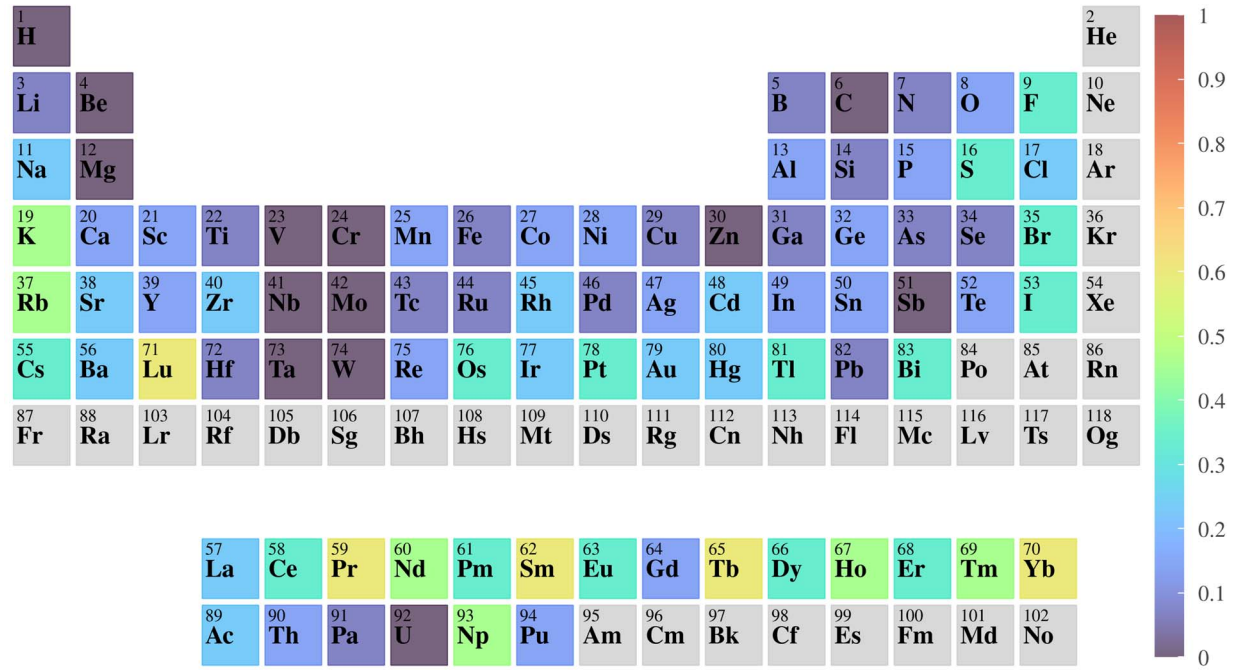
4. To calculate the similarity of representations of element i using embedding matrix **E** and feature matrix **H**, we calculate the overlap ratio of the set of top-k similar elements to i obtained from $S^{\mathbf{E}}$ and $S^{\mathbf{H}}$ which is $\frac{\left|\mathrm{Top}_k^{\mathbf{H}}(e) \cap \mathrm{Top}_k^{\mathbf{E}}(e)\right|}{\left|\mathrm{Top}_k^{\mathbf{H}}(e)\right|} = \frac{\left|\mathrm{Top}_k^{\mathbf{H}}(e) \cap \mathrm{Top}_k^{\mathbf{E}}(e)\right|}{k}$. By definition of this measure, 1.0 overlap ratio means $\mathrm{Top}_k^{\mathbf{H}}(e)$ and $\mathrm{Top}_k^{\mathbf{E}}(e)$ are the same 0 overlap ratio means these two sets are mutually exclusive.

It is important to note that while we employ the negative l_2 distance as a similarity metric for the feature matrix **H**, this measure is not asserted to possess a specific physical interpretation. Rather, we contend that the application of such metrics as a similarity measure effectively retrieves chemically similar elements when considering the top-*k* nearest neighbors for each element. Given the practical inconvenience of enumerating the complete set of top-*k* closest neighbors for the embedding matrix **E**, we utilize this similarity measure only as a proxy to illustrate that element embeddings do not inherently encode chemical relationships, and this analysis serves primarily for visualization purposes.

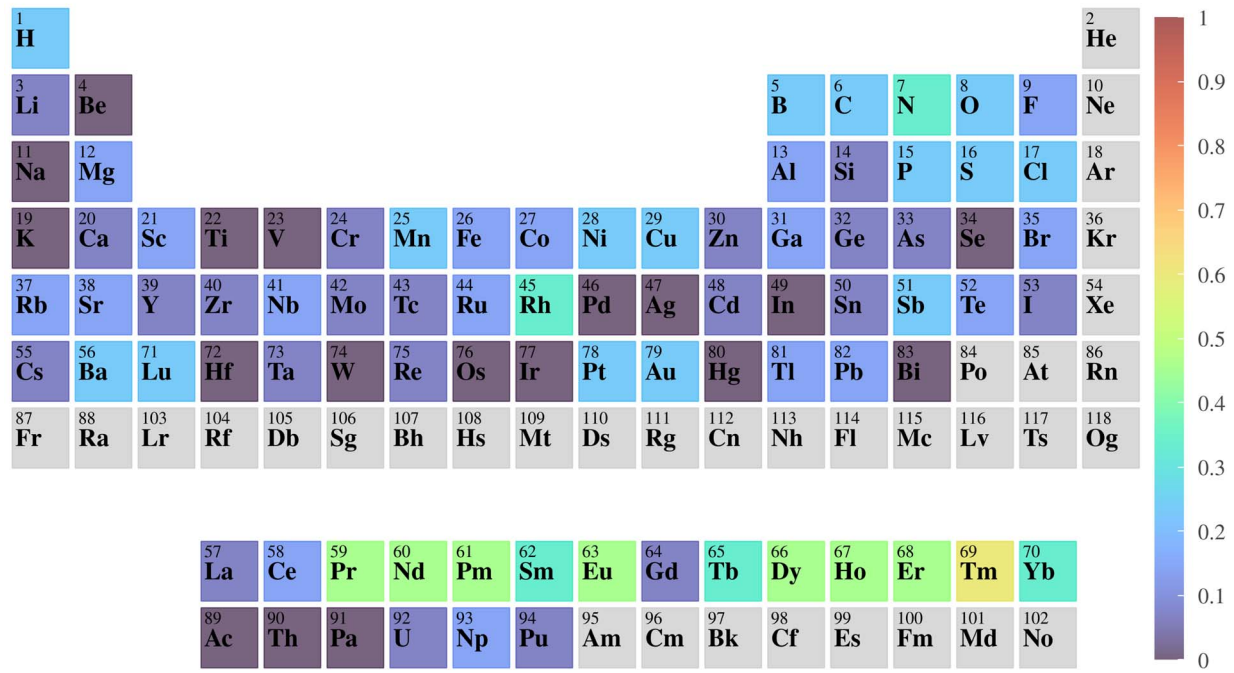
We have visualized the obtained the overlap ratio of $\mathrm{Top}_k^{\mathbf{H}}(e)$ and $\mathrm{Top}_k^{\mathbf{E}}(e)$ for $k = 8$ using a heat map over the periodic table in Fig. 3 for MACE and SchNet. Fig. 3 shows that the obtained overlap ratios are generally small for most of the elements. There are some elements in both heatmaps Fig. 3a and b that show a larger overlap ratio, but it is not true for the majority of the elements in the dataset. This trend was consistent across different similarity measures and reasonable choices of k. From this analysis we can conclude that the embeddings from matrix **E** are not similar to the feature matrix **H** and likely do not contain significant chemical correlations.

Probing chemical information in element embedding vectors

Similar to word embeddings, if element embeddings capture chemical information, their relative positions should reflect this chemistry. So different with the previous section where we compared **E** with a reference feature matrix **H**, we are going to probe the relative positions of element embeddings within **E**. If the embeddings merely serve as identifiers for each element,



(a) MACE



(b) SchNet

Fig. 3 Heatmaps of overlap ratios of top 8 similar elements for E and H. Smaller overlap ratio for a specific element shows that the corresponding embedding vector contains less chemistry.

their relative positions will not convey meaningful information. To determine whether the learned embeddings contain useful information, we test the following hypothesis: If the embeddings encode chemical information, the vector for an unseen element during initial training should be recoverable through further training on new data containing compounds with that element. This is because the approximate position of the unseen element should be inferred from the relative positions of other elements' embedding vectors. Thus, it should be possible to optimize the embedding of an unseen element post-training, improving the model's performance for those previously unseen elements.

We conducted tests on several elements for both MACE and SchNet using the following scenario. Firstly, we removed the compounds containing element e from the training set and optimized a model on $\mathscr{D}^{\text{Train}}_{-\{e\}}$. Secondly, we kept all the samples removed in the first phase, denoted by $\mathscr{D}^{\text{Train}}_{\{e\}}$, and disabled gradients for all parameters and the embedding matrix $\mathbf{E}$ except the corresponding row for e. Then, we trained this embedding vector using the complete dataset $\mathscr{D}^{\text{Train}}$. Despite updating the embedding vector for e, we found that the performance of the model on $\mathscr{D}^{\text{Test}}_{\{e\}}$ did not improve at all. To confirm that this observation was not due to an optimization issue, we deliberately corrupted the embedding vector for a chosen element e after training the model on the complete dataset $\mathscr{D}^{\text{Train}}$, and checked if we could recover the performance by optimizing the embedding vector for e. We added noise to the embedding vector e, which significantly degraded the model's performance on $\mathscr{D}^{\text{Test}}_{\{e\}}$. We observed that we could recover the performance using gradient descent or Adam after adding noise to the embedding vector for e, so that optimizing the embedding vector for e was not the issue.

These two observations demonstrate that the learned embedding matrix $\mathbf{E}$ does not encode much information about elements and that the embeddings are probably just indicators for elements.

Proposed method

We just discussed how learned embeddings do not embed significant chemical similarity between elements and probably serve as mere indicators of the element they represent. Therefore, it seems reasonable to suppose that we could use different embeddings that not only indicate species but also have chemical correlations, potentially gaining capabilities for the model. To explore this, instead of using an independent embedding matrix $\mathbf{E}$, we assume that the model has access to the feature matrix of elemental properties $\mathbf{H}$, where the i-th row of $\mathbf{H}$ represents the feature vector for the element with atomic number i. As shown in Fig. 4, to utilize $\mathbf{H}$, we employ a fully connected neural network, denoted by M_Θ and parameterized by Θ, that takes the feature matrix $\mathbf{H}$ (which is 94×58 dimensional as defined before) as input and outputs a d-dimensional representation for each element, where d is an adjustable hyper-parameter of a model. With this modification, our model becomes $f_\theta(\mathbf{z}_i, \mathbf{x}i, M\Theta(\mathbf{H}))$, and θ and Θ are optimized during training. Through this modification, the model learns to characterize each element based on its meaningful chemical features. Thus, even if we lack data samples for certain elements during training, we can still make reasonable predictions based on their elemental features, as there are similarities between different elements.

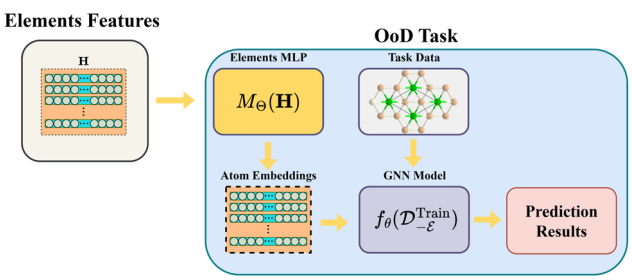


Fig. 4 The layout of our proposed method for using elements information in potential energy estimation.

Experiments

Task data generation

To demonstrate the efficacy of our proposed model, we conducted a series of experiments. We randomly choose a set of

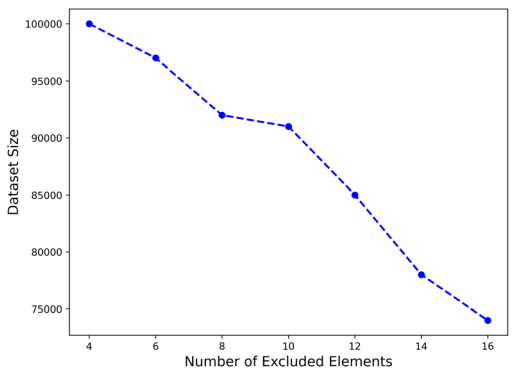


Fig. 5 This figure illustrates the relationship between the training data size and the number of elements for the defined OoD task. The number of excluded samples, as well as the rate at which samples are lost from the base dataset, increases as more elements are removed. Consequently, our proposed method cannot maintain its performance once more than 10 elements are excluded. This is likely because, beyond this threshold, the removal of key elements limits the model's ability to generalize to unseen elements.

Table 3 Performance comparison of the original MACE and SchNet architectures trained on full data and $\mathscr{D}^{\text{Train}}_{-\mathscr{E}}$, using both the elements MLP and the original models, evaluated with the MAE (eV per atom) metric. Our method provides a substantial improvement over the original models trained on the OoD training data $\mathscr{D}^{\text{Train}}_{-\mathscr{E}}$.

Excluded elements	Model	Full data	Elements MLP	OoD data (last epoch)	OoD data (best epoch)
$\mathscr{E}_1$	MACE	0.0190	0.1076	0.5560	0.43
	SchNet	0.0760	0.1325	0.3210	0.31
$\mathscr{E}_2$	MACE	0.0183	0.0874	0.5906	0.426
	SchNet	0.0695	0.1247	0.5103	0.443
$\mathscr{E}_3$	MACE	0.0207	0.0693	0.5302	0.372
	SchNet	0.0793	0.1201	0.5187	0.455

elements $\mathscr{E}$, then we remove any compound in the training set that contains any member of $\mathscr{E}$. For the test set, we keep every compound that contains at least one element in $\mathscr{E}$. We divide the data into a training set (85%) and a test set (15%) and use the same fixed split to generate $\mathscr{D}^{\text{Train}}_{-\mathscr{E}}$ and $\mathscr{D}^{\text{Test}}_{\mathscr{E}}$. We also refer to the subset of compounds in $\mathscr{D}^{\text{Test}}$ not containing elements in $\mathscr{E}$ with $\mathscr{D}^{\text{Test}}_{-\mathscr{E}}$.

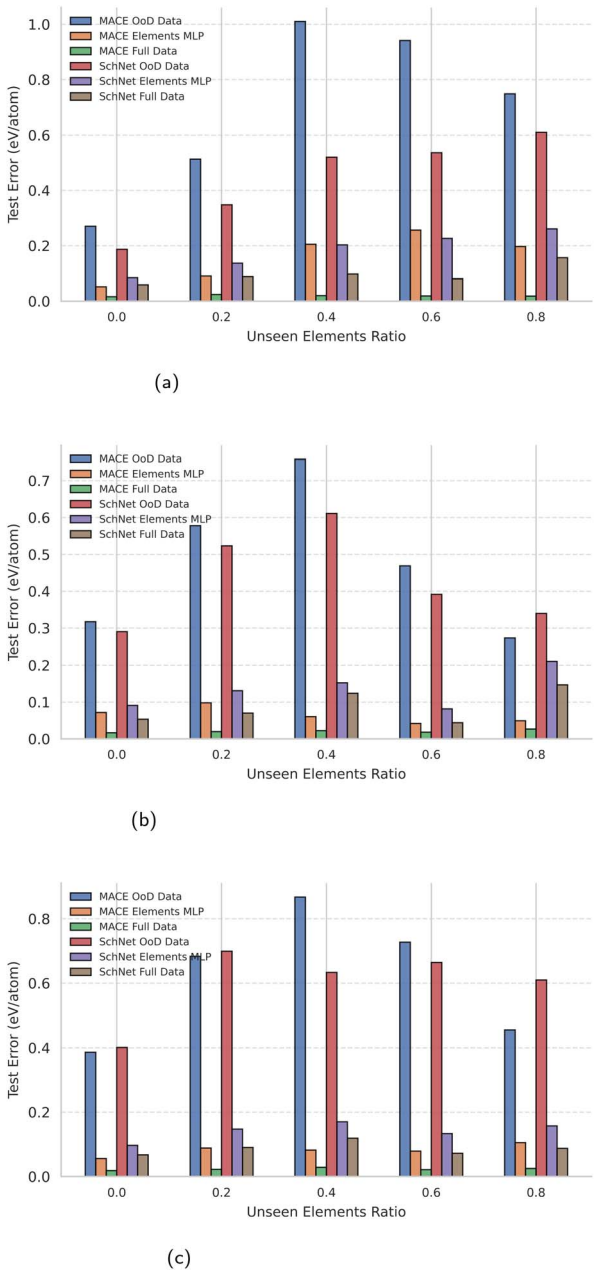


Fig. 6 Compounds in the dataset possessing at least one previously unseen element were grouped by the ratio of unseen elements they consist of. The excluded elements for each figure are mentioned in the corresponding caption. We plotted the MAE/Atom error against this ratio in each group. As anticipated, our model's performance is significantly superior to that of the model trained on OoD training data. (a) Na, Se, Br, Y, Ru, Sb, Tb, Ac. (b) Sc, V, Cu, Sr, Tb, Ho, Tm, Yb, Np. (c) K, Ca, Ti, Cd, Te, Er, Ir, U.

Baselines and our model

We use MACE and SchNet for f_θ and a simple two-layer MLP with SwiGlu activations[39] for M_Θ. We call our proposed models Elements MLP MACE and Elements MLP SchNet. To measure the impact of the missing data on generalization in the outlined OoD scenario, we use the original MACE and SchNet trained on the complete dataset and $\mathscr{D}^{\text{Train}}_{-\mathscr{E}}$ for each set of excluded elements $\mathscr{E}_i$ for $i \in \{1, 2, 3\}$ as baselines, which we refer to as "MACE Full Data", "SchNet Full Data", "MACE OoD Data" and "SchNet OoD Data".

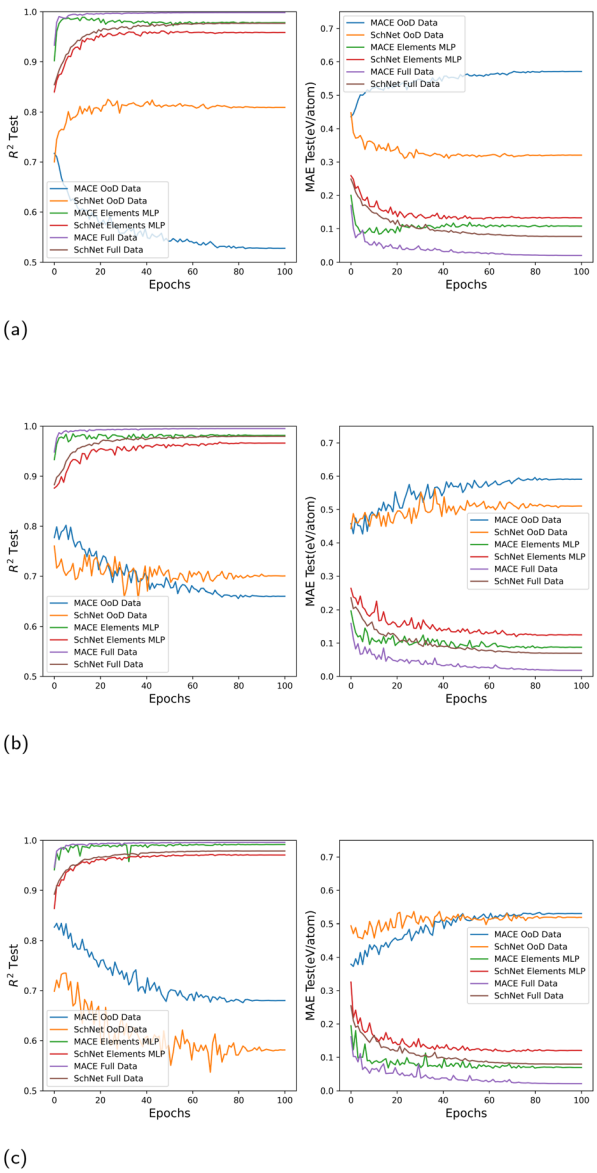


Fig. 7 The performance of the model that uses element features remains comparable to the model trained on the complete dataset when we randomly remove several elements from the dataset. (a) Models R^2 and MAE for the test set when samples that include Na, Se, Br, Y, Ru, Sb, Tb, Ac are removed from the training set. (b) Models R^2 and MAE for the test set when samples that include Sc, V, Cu, Sr, Tb, Ho, Tm, Yb, Np are removed from the training set. (c) Models R^2 and MAE for the test set when samples that include K, Ca, Ti, Cd, Te, Er, Ir, U are removed from the training set.

Formation energy prediction

We compare the performance of our proposed model against SchNet and MACE using three randomly sampled sets of excluded elements, each with different members, as detailed below.

- $\mathscr{E}_1 = \{\text{Na, Se, Br, Y, Ru, Sb, Tb, Ac}\}$
- $\mathscr{E}_2 = \{\text{Sc, V, Cu, Sr, Tb, Ho, Tm, Yb, Np}\}$
- $\mathscr{E}_3 = \{\text{K, Ca, Ti, Cd, Te, Er, Ir, U}\}$

It is important to note that removing specific elements from the dataset to create OoD tasks reduces the size of the training set. For example, excluding Cu and Co might result in the loss of 5000 samples containing these elements. To ensure a fair comparison, we adjusted the number of epochs for the complete dataset, ensuring both models underwent the same total number of optimization (gradient update) steps.

The performance comparison between our proposed method and the models with access to full training data and OoD data is represented in Table 3 for each experiment. We use the test set MAE from the last training epoch and the best-performing epoch as the comparison metric. Our method achieves a substantial improvement over the original models trained on the OoD training data $\mathscr{D}^{\text{Train}}_{-\mathscr{E}}$, while its performance remains lower to that of the model with full dataset access. We have observed consistent results for other randomly sampled sets up to $k = 10$, where k is the number of removed elements in the set. However, beyond this threshold, the gap between the models becomes larger with a higher rate. This is illustrated in Fig. 5. As we remove more than ten elements from the dataset, we lose data at a higher rate when the size of the set of the removed elements is greater than 10.

The R^2 and MAE values for the test set predictions were plotted in Fig. 7 as a function of training epochs for $\mathscr{E}_1$, $\mathscr{E}_2$ and $\mathscr{E}_3$. The results presented in Fig. 7 show that our models outperform the original models trained on OoD data by a large margin, even with several elements excluded from the training. This indicates that by leveraging element features, our models maintain generalization to unseen elements, despite the complete removal of all compounds containing the excluded set $\mathscr{E}$ from the training data. Notably, and consistent with the findings of Li *et al.*,[26] the performance of the original models trained on OoD data can worsen with more training. This negative effect is less strong for the Elements MLP models and is only seen slightly for the Elements MLP MACE models, as shown in Fig. 7a and b.

Performance *versus* the Ratio of Unseen Elements. We also examined the performance of our model across different ratios of unseen elements within compounds. To assess this, we categorized the compounds including at least one unseen element in the dataset based on the stoichiometry of unseen elements they contained. Then we rounded up ratios for each of the compounds and classified them in 0.0–0.2, 0.2–0.4, …, 0.8–1.0 groups. For example, if W was unseen and O seen, then WO3

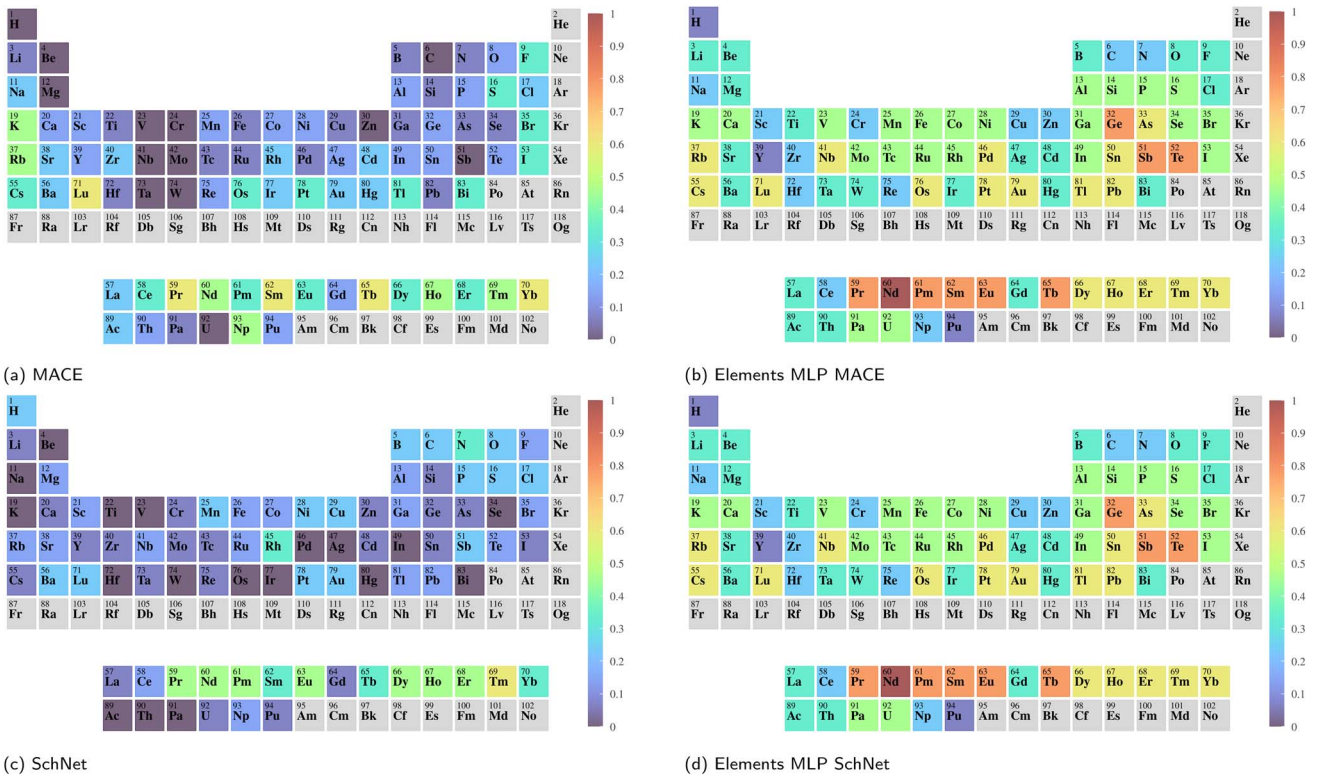


Fig. 8 Similarity heatmaps illustrating the overlap ratios of the top 8 most similar elements, as determined by the embedding matrices **E** of the original MACE and SchNet models (left column) and their corresponding Elements MLP augmented versions (right column), compared to the element feature matrix **H**. Higher overlap ratios in the right column indicate that the elements MLP embeddings capture more of the inherent chemical information present in **H**.

would have ration 0.25 of unseen to seen. Notice that even compounds in the 0.0–0.2 group contain at least one unseen elements. Fig. 6 illustrates the performance of our models compared to the original models with access to the complete and OoD dataset. As depicted in Fig. 6, our proposed method consistently shows substantially better performance than models trained on OoD data, while still performing reasonably worse than models trained on the full training data. One might expect that the performance gap between both our models and the OoD models, relative to the models with full data access, would widen as the proportion of unseen elements increases. However, Fig. 6 does not show such a trend. We must be careful to note that the error is measured using a metric already normalized by the number of atoms. Therefore, even if the ratio of unseen elements increases, this effect is offset by the division by the number of atoms.

Similarity between elements MLP output and H

One might wonder if the embedding matrix from the Elements MLP would resemble **H**, following the same method used to visualize the similarity heatmap between the original models' embedding matrix **E** and **H**. We've plotted this comparison in Fig. 8. As the figure shows, the Elements MLP models' element embedding matrices are considerably more similar to **H** than those of the original models.

Conclusion and discussion

This work addresses a challenge in Graph Neural Networks for potential energy prediction: generalizing to out-of-distribution scenarios with unseen elements. Through our experiments, we show that the common practice of using element embeddings leads to them lacking meaningful chemical correlations and likely acting as mere identifiers. This limitation becomes evident when elements absent from the training data are encountered, leading to degraded performance. To address this, we feed elemental features—such as atomic size and electronegativity—directly into the model, improving generalization for unseen elements. Even when multiple unseen elements are introduced simultaneously, our model maintains reasonable accuracy with modest degradation compared to models trained with full datasets.

While the feature-based approach enhances generalization in OoD settings, models trained solely with elemental features show only marginal improvements on IID test data. Although we did not observe significant gains in the IID setting, using chemically meaningful embeddings may allow for smoother adaptation when new elements are introduced. Additionally, this approach could enable better transfer learning from existing datasets, further boosting performance. Incorporating more advanced chemical features and uncertainty quantification could make these models more robust and scalable for real-world applications. However, our experiments also reveal a persistent performance gap between models trained with full data access and those handling compounds with unseen elements. Reducing this gap is an important avenue for future research.

In conclusion, our study demonstrates that incorporating elemental properties as features improves out-of-distribution generalization when predicting the potential energies of compounds from their atomic structures. Although unexplored in this project, we hypothesize that our method could extend to any other molecular target or property that depends on the type and properties of the constituent elements. While these features equip the model to handle compounds with unseen elements, a notable gap remains compared to models trained with complete data. Closing this gap will require further investigation, possibly through modifications to the loss function or model architecture in future work.

Author contributions

Conceptualisation, methodology, software, data curation, formal analysis and original draft preparation: H. M. Supervision, project administration and funding acquisition: V. H. and D. M. Writing– review & editing: all authors. All authors have read and approved the final manuscript.

Conflicts of interest

There are no conflicts to declare.

Data availability

Our experiments utilized the Matbench v0.1 test suite, which is publicly accessible through the Matminer Python library. The complete implementation of the experiments, including the code, relevant scripts, and our data is available in the paper's Mendeley repository.[40]

Supplementary information is available. See DOI: **https://doi.org/10.1039/d5dd00182j**.

Acknowledgements

Support for H. M. and V. H. was provided by the National Science Foundation under NSF Award 2020243 “AI Institute: Planning: Institute for AI Enabled Materials Discovery, Design, and Synthesis”. Support for D. M. was provided by the National Science Foundation under NSF Award Number 1931298 “Collaborative Research: Framework: Machine Learning Materials Innovation Infrastructure”.

Notes and references

1 B. Leimkuhler and C. Matthews, *Molecular Dynamics*, Springer International Publishing, 2015, p. 5.
2 P. Reiser, M. Neubert, A. Eberhard, L. Torresi, C. Zhou, C. Shao, H. Metni, C. van Hoesel, H. Schopmans, T. Sommer and P. Friederich, *Commun. Mater.*, 2022, **3**, 93.
3 A. K. A. Kandy, K. Rossi, A. Raulin-Foissac, G. Laurens and J. Lam, *Phys. Rev. B*, 2023, **107**, 174106.

4 J. Gasteiger, F. Becker and S. Günnemann, *Adv. Neural Inf. Process Syst.*, 2021, 3.
5 Y. Liu, L. Wang, M. Liu, Y. Lin, X. Zhang, B. Oztekin and S. Ji, *International Conference on Learning Representations*, 2022.
6 J. Gasteiger, J. Groß and S. Günnemann, *International Conference on Learning Representations*, 2020.
7 K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko and K.-R. Müller, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.
8 S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt and B. Kozinsky, *Nat. Commun.*, 2022, **13**, 2453.
9 F. Fuchs, D. Worrall, V. Fischer and M. Welling, *Adv. Neural Inf. Process Syst.*, 2020, 1970–1981.
10 Y.-L. Liao and T. Smidt, *The Eleventh International Conference on Learning Representations*, 2023.
11 Y.-L. Liao, B. M. Wood, A. Das and T. Smidt, *The Twelfth International Conference on Learning Representations*, 2024.
12 V. G. Satorras, E. Hoogeboom and M. Welling, *Proceedings of the 38th International Conference on Machine Learning*, 2021, pp. 9323–9332.
13 J. Brandstetter, R. Hesselink, E. van der Pol, E. J. Bekkers and M. Welling, *International Conference on Learning Representations*, 2022.
14 S.-C. Li, H. Wu, A. Menon, K. A. Spiekermann, Y.-P. Li and W. H. Green, *J. Am. Chem. Soc.*, 2024, **146**, 23103–23120.
15 L.-Y. Chen, T.-W. Hsu, T.-C. Hsiung and Y.-P. Li, *J. Phys. Chem. A*, 2022, **126**, 7548–7556.
16 J. S. Smith, R. Zubatyuk, B. Nebgen, N. Lubbers, K. Barros, A. E. Roitberg, O. Isayev and S. Tretiak, *Sci. Data*, 2020, **7**, 134.
17 A. Zhu, S. Batzner, A. Musaelian and B. Kozinsky, *J. Chem. Phys.*, 2023, **158**, 164111.
18 A. R. Tan, S. Urata, S. Goldman, J. C. B. Dietschreit and R. Gómez-Bombarelli, *npj Comput. Mater.*, 2023, **9**, 225.
19 S. Thaler, G. Doehner and J. Zavadlav, *J. Chem. Theory Comput.*, 2023, **19**, 4520–4532.
20 Y. Li, W. Xiao and P. Wang, Uncertainty Quantification of Artificial Neural Network Based Machine Learning Potentials, *Proceedings of the ASME 2018 International Mechanical Engineering Congress and Exposition*, ASME, Pittsburgh, Pennsylvania, USA, 2018, vol. 12, p. V012T11A030.
21 C. Chen and S. P. Ong, *Nat. Comput. Sci.*, 2022, **2**, 718–728.
22 B. Deng, P. Zhong, K. Jun, J. Riebesell, K. Han, C. J. Bartel and G. Ceder, *Nat. Mach. Intell.*, 2023, **5**, 1031–1041.
23 I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, F. Berger, N. Bernstein, A. Bhowmik, S. M. Blau, V. Cărare, J. P. Darby, S. De, F. D. Pia, V. L. Deringer, R. Elijošius, Z. El-Machachi, F. Falcioni, E. Fako, A. C. Ferrari, A. Genreith-Schriever, J. George, R. E. A. Goodall, C. P. Grey, P. Grigorev, S. Han, W. Handley, H. H. Heenen, K. Hermansson, C. Holm, J. Jaafar, S. Hofmann, K. S. Jakob, H. Jung, V. Kapil, A. D. Kaplan, N. Karimitari, J. R. Kermode, N. Kroupa, J. Kullgren, M. C. Kuner, D. Kuryla, G. Liepuoniute, J. T. Margraf, I.-B. Magdău, A. Michaelides, J. H. Moore, A. A. Naik, S. P. Niblett, S. W. Norwood, N. O'Neill, C. Ortner, K. A. Persson, K. Reuter, A. S. Rosen, L. L. Schaaf, C. Schran, B. X. Shi, E. Sivonxay, T. K. Stenczel, V. Svahn, C. Sutton, T. D. Swinburne, J. Tilly, C. van der Oord, E. Varga-Umbrich, T. Vegge, M. Vondrák, Y. Wang, W. C. Witt, F. Zills and G. Csányi, *A foundation model for atomistic materials chemistry*, 2024, **https://arxiv.org/abs/2401.00096**.
24 K. Choudhary, B. DeCost, L. Major, K. Butler, J. Thiyagalingam and F. Tavazza, *Digital Discovery*, 2023, **2**, 346–355.
25 J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals and G. E. Dahl, *Proceedings of the 34th International Conference on Machine Learning*, 2017.
26 K. Li, A. N. Rubungo, X. Lei, D. Persaud, K. Choudhary, B. DeCost, A. B. Dieng and J. Hattrick-Simpers, *Commun. Mater.*, 2025, **6**, 9.
27 W. Hu, M. Shuaibi, A. Das, S. Goyal, A. Sriram, J. Leskovec, D. Parikh and C. L. Zitnick, *ForceNet: A Graph Neural Network for Large-Scale Quantum Calculations*, 2021.
28 L. Ward, A. Agrawal, A. Choudhary and C. Wolverton, *npj Comput. Mater.*, 2016, **2**, 16028.
29 A. Dunn, Q. Wang, A. Ganose, D. Dopp and A. Jain, *npj Comput. Mater.*, 2020, **6**, 138.
30 A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder and K. A. Persson, *APL Mater.*, 2013, **1**, 011002.
31 J. P. Perdew and W. Yue, *Phys. Rev. B:Condens. Matter Mater. Phys.*, 1986, **33**, 8800–8802.
32 I. Batatia, D. P. Kovacs, G. N. C. Simm, C. Ortner and G. Csanyi, *Adv. Neural Inf. Process Syst.*, 2022, 1–5.
33 M. Rahm, R. Hoffmann and N. W. Ashcroft, *Chem.–Eur. J.*, 2016, **22**, 14625–14632.
34 B. Cordero, V. Gómez, A. Platero-Prats, M. Revés, J. Echeverría, E. Cremades, F. Barragán and S. Álvarez, *Dalton Trans.*, 2008, 2832–2838.
35 P. Pyykkö and M. Atsumi, *Chem.–Eur. J.*, 2009, **15**, 186–197.
36 P. Pyykkö and M. Atsumi, *Chem.–Eur. J.*, 2009, **15**, 12770–12779.
37 P. Pyykkö, S. Riedel and M. Patzschke, *Chem.–Eur. J.*, 2005, **11**, 3511–3520.
38 T. Mikolov, K. Chen, G. S. Corrado and J. Dean, *International Conference on Learning Representations*, 2013.
39 N. Shazeer, *CoRR*, 2020, abs/2002.05202.
40 H. Mahdavi, D. Morgan and V. Honavar, *Beyond Training Data: How Elemental Features Enhance ML-Based Formation Energy Predictions (Version V2) Mendeley Data.*, 2025, DOI: **10.17632/n3cwj2hb7w.2**.