Graph Representation of Local Environments for Learning High-Entropy Alloy Properties

Hengrui Zhang
Northwestern University
hrzhang@u.northwestern.edu

Ruishu Huang*
University of Wisconsin–Madison
rhuang255@wisc.edu

Jie Chen*
Virginia Tech
jiechen@vt.edu

James M. Rondinelli
Northwestern University
jrondinelli@northwestern.edu

Wei Chen
Northwestern University
weichen@northwestern.edu

Abstract

Graph neural networks (GNNs) have excelled in predictive modeling for both crystals and molecules, owing to the expressiveness of graph representations. High-entropy alloys (HEAs), however, lack chemical long-range order, limiting the applicability of current graph representations. To overcome this challenge, we propose a representation of HEAs as a collection of local environment (LE) graphs. Based on this representation, we introduce the LESets machine learning model, an accurate, interpretable GNN for HEA property prediction. We demonstrate the accuracy of LESets in modeling the mechanical properties of quaternary HEAs. Through analyses and interpretation, we further extract insights into the modeling and design of HEAs. In a broader sense, LESets extends the potential applicability of GNNs to disordered materials with combinatorial complexity formed by diverse constituents and their flexible configurations.

1 Introduction

Advances in machine learning (ML) have greatly facilitated the understanding and design of materials. In particular, graph neural networks (GNNs) have undergone extensive development and shown outstanding performance across various tasks for both crystalline and molecular materials [1], mainly benefiting from the versatility of graph representation in capturing local chemical environments. For molecules, a graph representation is straightforwardly defined by encoding atoms as nodes and chemical bonds as edges [2]. This method has then been adapted to crystals [3], whose repeating units are represented as graphs, with atoms as nodes and nearest neighbors connected by edges. Moreover, graphs are flexible to incorporate physics information, such as electrical charge [4], spin [5], or symmetry [6], which is highly desirable for materials science applications.

Adopting GNNs to high-entropy alloys (HEAs) has been deemed challenging [7]. Single-phase HEAs are chemically disordered alloys formed by multiple metal elements often with near-equimolar concentrations that can lead to short-range correlations [8]. They form a promising candidate space for materials with desired properties ranging from thermal [9], mechanical [10] to catalysis [11]. Owing to the lack of periodic (long-range) chemical order of the multiple species, atomistic representations of HEAs with graphs are not as easy to construct as they are for molecules or crystals. Existing ML models for HEAs mainly use tabular descriptors [12–14] or graph representation of the composition [15], whereas local environments are not considered.

*Work performed while at Northwestern University.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

Our recent work [16] proposes a "graph set" representation for molecular mixtures, along with an ML model architecture named MolSets to learn molecular chemistry and make predictions of mixture properties. We recognize that HEAs have similar characteristics as molecular mixtures: both are unordered collections of local chemical environments (molecules or atomic sites). With this analogy, we propose a graph set representation of HEAs and an extended ML model LESets to predict the properties of HEAs represented thereby. The proposed representation and model architecture extend the applicability of GNNs to HEAs and other disordered materials, so that the modeling of these materials systems can benefit from the rapid development of graph-based ML models. The remaining sections are organized as follows: Section 2 reviews the literature of related works. In Section 3, we describe the proposed representation and model. We then present results including benchmark, sensitivity analysis, and model interpretation in Section 4. Section 5 discusses the benefits, outlook, and limitations.

2 Related Works

In this section, we briefly review the literature in two relevant directions: (1) GNNs for materials property prediction and (2) computational modeling of HEAs at the atomic scale.

GNNs for materials. GNNs have become an essential method in materials structure–property modeling. The key to building such models is to represent the structure of a material as a graph, a data structure with nodes and edges. Intuitively, graphs are formulated with atoms as nodes, and atoms that share electrons (bonded or are nearest neighbors) are connected by edges, with atomic/bonding descriptors as node/edge features. On graph data, GNNs perform "message passing" to fuse node/edge information and predict graph-level outputs, such as materials properties. Following this approach, GNNs have been applied to molecules [2] and crystals [3]. Yet, the applicability is limited to materials with well-defined structures or repeating units, whereas for unordered, combinatorial materials systems, such as HEAs, a graph representation is hard to formulate.

Beyond the simple formulation, graph representation and GNNs are flexible to incorporate various physical information or knowledge. Chemprop [17] employs bi-directional message passing to attain high accuracy in molecular property prediction. ALIGNN [18] represents bonds and angles by nodes and edges, respectively, thus incorporating angular information in crystals. M3GNet [19] and MACE [20] utilize many-body interactions; CHGNet [4] and SpinGNN [5] encode electrical charge and spin on atoms, respectively. PotNet [21] explicitly includes the physical forms of interatomic potentials. Another broad category of physical information is symmetry, which is incorporated in GNNs in the form of invariance or equivariance [6, 22]. In addition, model form innovations are explored and brought to materials science, leading to transformer-based GNNs such as Matformer [23]. As the method proposed in this work is a generic framework agnostic of the exact form of GNN, these are orthogonal yet complementary directions that could be combined with the framework we propose.

Modeling HEAs. One key challenge to the computational modeling of HEAs is the randomness, or lack of order, in their structures. The atomic scale physics-based modeling techniques, such as density functional theory and molecular dynamics, require well-defined structures as input. Due to the lack of order in HEAs, such studies generally use (special quasi) random structures given composition, construct large supercells to account for the randomness, or construct cluster expansions [24–26].

Machine learning models take various approaches to handle or bypass such randomness. One way is to form a tabular input data structure from summary statistics of elemental descriptors and model it using conventional ML methods, such as gradient boosting and random forest [27], or neural networks. Other ways include constructing input representations from multiple random realizations of structures or frequency counting [28]. There are also GNN methods, such as ECNet [15], working on graph representations of composition alone. However, the capability of GNNs to capture connectivity and interactions in local chemical environments has not yet been unleashed in HEAs.

3 Methods

3.1 Representing HEA by local environments

As Figure 1(a) shows, a typical HEA structure is unordered at the atomic scale. Specifically, the atomic sites in the structure may be fixed, but the arrangement (or decoration) of atoms among those sites is nearly random. To formulate a graph representation, we focus on local environments (LEs), i.e., an atom and its nearest neighbors. For an atom (e.g., Fe), its neighbors may be all types of elements, appearing at frequencies determined by their molar fractions. We represent this local environment by a graph, where the Fe atom is the center node connected to all other types of atoms, whose molar fractions are the edge weights. Note that though the Fe node is not directly connected to itself, its information could propagate to itself via message passing. Each node is associated with a feature vector of elemental descriptors: one-hot encoded elemental period and group, atomic mass (Dalton), covalent radius (Å), Pauling electronegativity, first ionization energy (eV), electron affinity (eV), and atomic volume ($\mathrm{cm}^3\,\mathrm{mol}^{-1}$). The weight fraction of a non-center atom is used as an edge feature. Every type of LE is thus represented as a graph.

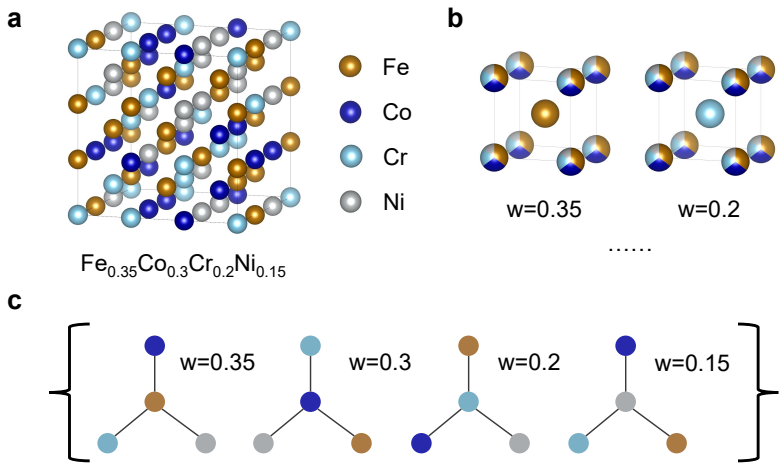

Figure 1: (a) Illustration of the atomic-scale structure of a quaternary HEA. (b) The local environments. (c) Representing the HEA as a set of local environment graphs.

The HEA as a whole is an unordered collection of different local environments. For example, the quaternary HEA shown here consists of four types of LEs, centered by Fe, Co, Cr, and Ni, respectively. We represent the HEA as a set of four LE graphs x, each associated with a weight fraction w that is the fraction of the center element in the HEA's composition, formally, $X = \{(x_i, w_i)\}$. This graph set representation captures the atomic interactions of LEs, as well as the flexibility of relations between LEs, matching the unordered configuration of LEs in the HEA structure.

3.2 LESets model

In [16], we present MolSets that integrates GNN with the permutation invariant Deep Sets architecture [29] and attention mechanism [30] for molecular mixture property modeling. In MolSets, each molecule is represented as a graph, and a mixture is represented as a weighted set of graphs. Noticing the analogy of this data structure to the LE graph-based representation of HEAs, we extend MolSets to HEAs, named LESets.

LESets consists of three modules, as shown in Figure 2. Taking an input datapoint $X = \{(x_i, w_i)\}$, a GNN module ϕ maps each local environment graph x_i to its intermediate representation vector z_i, i.e., $z_i = \phi(x_i)$. Then, $\{(z_i, w_i)\}$ is aggregated into a global representation vector Z, which is mapped to the target response y (HEA property) by a multilayer perceptron ρ. Overall, the model learns an input–response function of the following form:

$$y = \rho\left(\oplus_{(x,w)\in X}\{\phi(x), w\}\right). \tag{1}$$

Specifically, for the aggregation $\oplus$, we take two approaches:

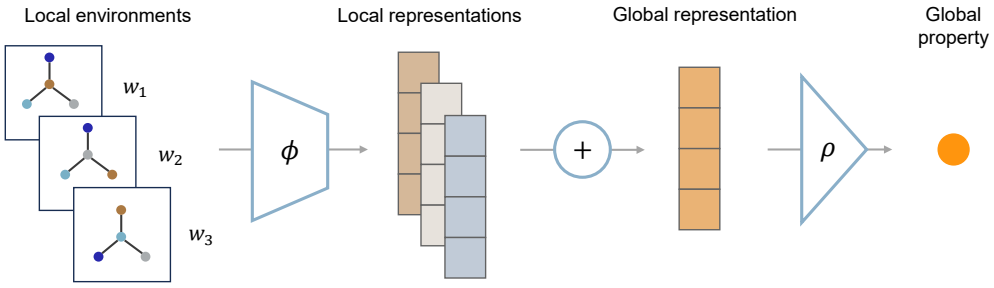

Figure 2: Architecture of LESets: A graph neural network ϕ learns the representations of local environments. Aggregation module $\oplus$ combines the local environment representations and their weights to form a global representation of the HEA. A multilayer perceptron ρ maps the global representation to target materials property.

1. Weighted summation of LE representations, i.e.,
$$\oplus\{\phi(x_i), w_i\} = \sum w_i \phi(x_i); \tag{2}$$
2. Adjust LE representations using attention mechanism:
$$z_i = \texttt{attention}[\phi(x_i)] \text{ for every } x_i \text{ in } X, \tag{3}$$
taking into account the disproportionate importance and interactions between LEs within a HEA, before weighted summation $\oplus\{z_i, w_i\} = \sum w_i z_i$.

We name the model using each approach "LESets-att" and "LESets-ws", respectively. A formal description of the model is provided in Algorithm 1 within Appendix A.

4 Results

4.1 Data and experimental setup

To test and demonstrate the performance of the LESets model, we conduct experiments on a previously reported DFT-calculated HEA property dataset [31]. The dataset contains bulk modulus B and Wigner-Seitz radius r_{ws} for 7086 HEAs, 1909 of which also have Young's modulus E reported. We test our and other ML models in predicting these properties and measure their performances using two metrics, coefficient of determination R^2 and mean absolute error (MAE).

We compare the LESets model with several previous methods: (1) The Deep Sets model with HEAs represented as sets of elements, as presented in [31]. Note that the major distinction of LESets from this model is incorporating local environment information encoded in graphs. (2) Conventional statistical learning methods, including gradient boosting decision trees, random forest, k-nearest neighbors (kNN), support vector machine (SVM), Lasso, and ridge regression, using elemental property descriptors, all implemented in `sklearn` [32]. For LESets, we also investigate the effect of using attention in the $\oplus$ module.

Experiments of the deep learning models (Deep Sets and LESets) are conducted with the following procedure. First, the hyperparameters of every model and target property are tuned using grid search on one random split of the dataset. The optimal hyperparameters found by tuning are listed in Table 1 within Appendix B. Then, the dataset undergoes 30 random splits into training, validation, and test data in a 3:1:1 ratio. In each split, the model is trained by minimizing mean squared error (MSE) loss on the training dataset using the AdamW optimizer [33], with a weight decay coefficient of 10^{-4} for regularization. The model's loss on the validation dataset is monitored for learning rate scheduling (learning rate is reduced by half after 10 epochs of no improvement) and early stopping (training is terminated after 20 epochs of no improvement). After training, the model makes predictions on the testing dataset, and its performance metrics are recorded. The models are implemented in PyTorch and PyTorch Geometric (PyG).

For the conventional ML models, experiments are conducted similarly: the hyperparameters are tuned using grid search and 5-fold cross-validation on 40% of the dataset, then their performances

are assessed on 30 random splits of the remaining data into training and testing in a 2:1 ratio. Computational details are provided in Appendix A.

4.2 Benchmark test results

In Figure 3, we show the performance metrics on predicting bulk modulus B and Young's modulus E for LESets, Deep Sets, and gradient boosting, which is the best among conventional ML models. As all models attain high R^2 and low error for r_{ws}, we exclude it from comparison here. We list the average performance metrics of all tested models and properties in Table 2 within Appendix B. To visualize different models' predictions, we show their predicted values vs. true values in one of the random data splits in Figure 4.

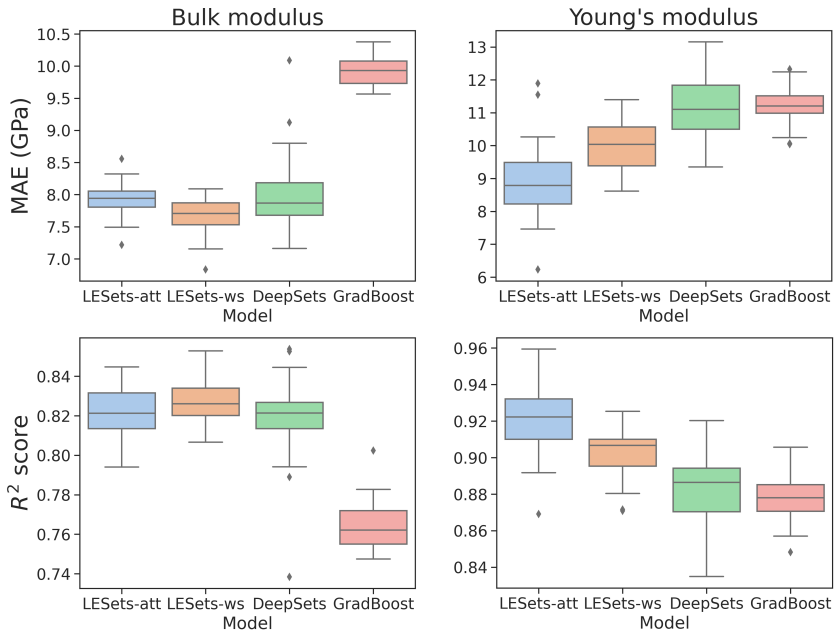

Figure 3: MAE and R^2 of LESets, Deep Sets, and gradient boosting on the testing dataset across 30 random data splits.

From the results, we find that LESets attains higher prediction accuracy than its counterparts. Notably, compared to Deep Sets, R^2 and MAE of LESets vary less across random data splits, which indicates better robustness to data variations. The improvement in accuracy could be because local chemical environment information, such as bonding, plays an important role in determining elastic properties [34].

A more interesting comparison is between two versions of LESets with different aggregation mechanisms. The simpler LESets-ws outperforms LESets-att in predicting B, suggesting that attention-based aggregation is unnecessary in this task. Whereas in predicting E, LESets-att regains the advantage. This difference indicates that elements or local environments may have disproportionate importance in determining E but not for B. Further investigations using electronic structure methods could find physical insights and guide the design principles of HEAs' elastic properties.

4.3 Sensitivity analysis

As a deep learning model, LESets requires sufficient data to be trained. Although high-throughput computation and data platforms have made materials data more available than before, understanding the data requirement guides model usage as well as data collection efforts [35]. To this end, we conduct sensitivity analyses of LESets' performance with respect to data size.

The analyses are conducted on the performant version of LESets on bulk and Young's moduli, respectively, viz. LESets-ws for B and LESets-att for E. For each property, we leave out 20% of the dataset to be the testing dataset and use a fraction of the remaining data in model training.

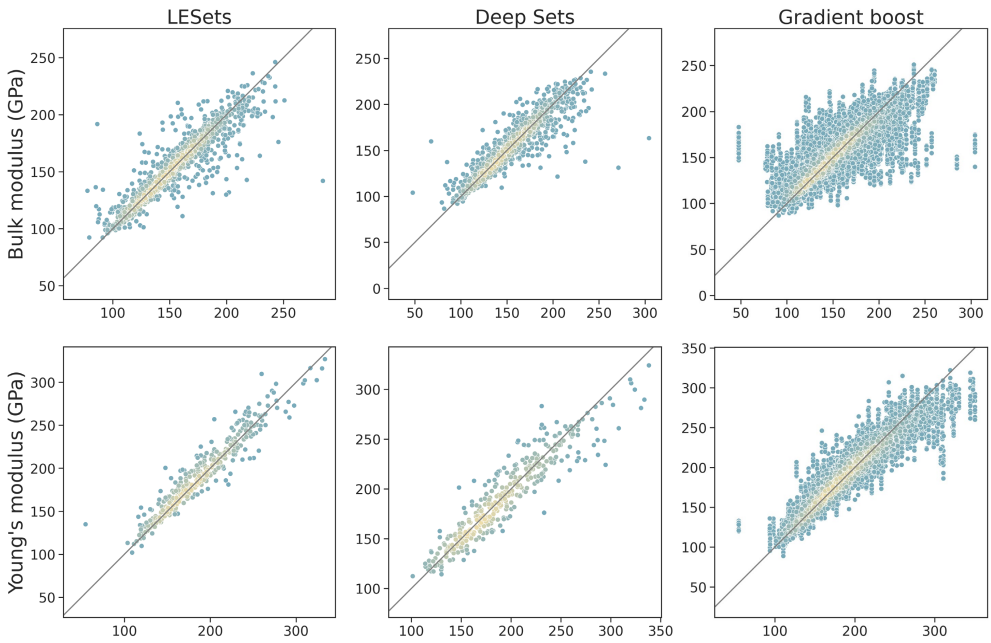

Figure 4: Regression plots showing target values (horizontal axis) and predicted values (vertical axis) of E and B for LESets (the better one between LESets-ws and LESets-att), Deep Sets, and gradient boosting. Colors reflect the density of points, quantified by kernel density estimation.

This fraction of data is split into training and validation datasets in a 3:1 ratio. We repeat this experiment 10 times with different random states for every fraction. Figure 5 shows the relation between performance metrics and data fraction. For B, LESets' performance begins saturating when the data fraction is high, whereas for E, it maintains a trend of increasing within the range of analysis. This can be partly explained by the much larger number of datapoints for B (7086) than that for E (1909). Note that LESets' number of trainable parameters typically ranges from 5000–8000 (varies depending on hyperparameters). The sensitivity analysis results suggest that the number of parameters might be a rough guideline for the required amount of data.

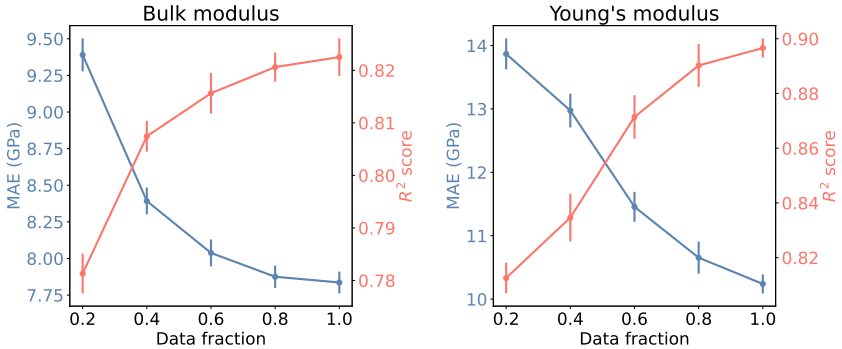

Figure 5: The change of LESets' MAE and R^2 on two target properties with data fraction. Points denote mean values across 10 replicates, and error bars denote standard error (standard deviation/$\sqrt{10}$).

4.4 Interpretation

One benefit of LESets-att is its interpretability provided by the attention-based aggregation. Following the approach in [16], we investigate the Euclidean norm change of a local environment's representation vector after attention:

$$Imp = \|z'\|_2 \,/\, \|z\|_2 \tag{4}$$

as a score that reflects its relative importance in the HEA. For simplicity, we refer to the importance of a local environment as the importance of its center element. With this importance score, we use two criteria to find out which elements contribute more to a HEA's Young's modulus E: (1) elements whose importance score is at least three times the lowest in the HEA; (2) elements with the largest score in the HEA. We plot the frequency of elements satisfying criterion (1) in Figure 6(a), and of those satisfying both criteria in Figure 6(b). The identified elements, such as Cr, Co, Mo, Hf, and W, appear to be frequent choices in previous studies designing high-stiffness HEAs [36].

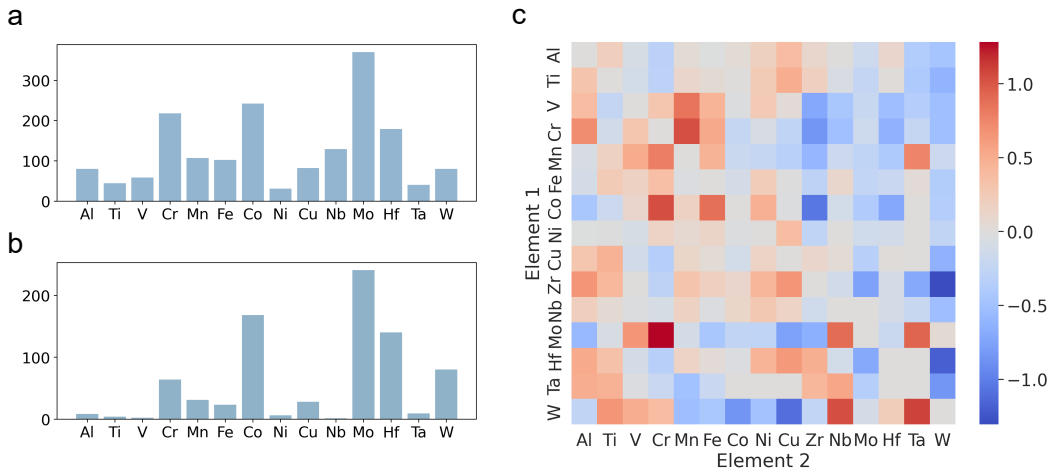

Figure 6: Frequency of (a) an element's Imp being at least $3\times$ the least important one, and (b) elements being the most important in a HEA while satisfying the previous criterion. Zr is excluded because of zero appearance. (c) Difference between Element 1's average Imp in all HEAs with and without Element 2.

Besides the importance of single elements, we are also interested in the interactions between elements. In Figure 6 (c), we show how an element's importance score is affected by the presence of another element. Though not identified as "important" by its own Imp, elements such as Mn, Nb, and Ta strongly correlate with other elements' contributions to E, and are thus beneficial as additives. Theories behind the additives' effects, such as mixing enthalpy, have been explored [37] and remain an active research topic. We envision these and further interpretations enabled by LESets can help derive useful design principles for HEAs and guide materials discovery.

5 Discussions

The methodological advances of LESets are two-fold. Firstly, the graph set representation effectively captures the chemistry of the local environment while maintaining flexibility to cope with the lack of long-range orders in HEAs. Secondly, the use of attention mechanism in aggregation enables finding disproportionate contributions of certain ingredients (LEs in this case). It also allows a qualitative assessment of the ingredients' importance. This interpretability is useful for guiding materials design and is desired for scientific ML models.

Note that the focus of this work is to highlight the applicability of graph representation and GNNs to HEAs, while accuracy is less emphasized. Hence, the accuracy of LESets shown in Section 4 potentially has room for improvement. Here, we suggest two possible approaches to further optimizing its accuracy: (1) Revisit the selection of elemental descriptors for constructing node features; (2) Fine-tune the layers' numbers and hidden dimensions, with a focus on trainable parameter numbers in each module of LESets.

The LESets model architecture has demonstrated general applicability for molecular mixtures in [16] and HEAs in this work. These are both instances of *combinatorial* materials systems, whose diverse constituents (LEs, molecules, etc.) and various configurations of constituents form an expansive space with multilevel complexity. Many other materials families, such as heteroanionic materials [38], heterostructures [39] and defected functional materials [40], are combinatorial materials systems or

possess combinatorial complexity. LESets can potentially provide a generic and extensible framework for ML method development for these materials systems.

Limitations. At last, we discuss the limitations of LESets and future work directions toward overcoming them. (1) As a demonstration, we use simple node, edge features, and GNN models in testing LESets. More informative features and advanced GNN model forms could further improve its predictive accuracy. (2) LESets' representation of local environment information is based on an assumption that all elements could appear in an atom's nearest neighbors, and simplified by modeling their appearance frequencies rather than exact locations. With emerging materials characterization techniques [41], the atomic-scale structures of HEAs can be obtained. When more exact structural data become available, LESets could utilize them to represent HEAs in a more informative way. (3) In addition to LEs and their configurations at the atomic scale, microstructure also impacts the mechanical properties of HEAs [42]. LESets currently does not take into account microstructure. A future work direction toward more accurate modeling of HEAs' mechanical properties is extending the representation and model to incorporate multiscale information.

6 Conclusion

In this work, we present a representation of high-entropy alloys (HEAs) as a collection of local environments, and the LESets machine learning model to accurately predict HEAs' properties using this representation. Through benchmark tests, we demonstrate the advantage of LESets in predicting HEA mechanical properties over existing ML models. Furthermore, we conduct sensitivity analyses to provide guidelines for LESets' data requirements. Finally, utilizing LESets' interpretability, we investigate elemental contributions to Young's modulus of HEAs, deriving insights to guide HEA design. The code is available at https://github.com/Henrium/LESets.

Acknowledgments and Disclosure of Funding

This work was supported in part by the U.S. National Science Foundation (Awards DMR-2324173 and DMR-2219489). H.Z. was also supported by the Ryan Graduate Fellowship. J.M.R. acknowledges support from the U.S. Department of Navy, Office of Naval Research (Award N000014-16-12280). The authors thank Wangshu Zheng, Xiao-Yan Li, and Pengfei Ou for their valuable insights.

References

1. Reiser, P. *et al.* Graph neural networks for materials science and chemistry. *Communications Materials* **3,** 93 (2022).
2. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. *Neural Message Passing for Quantum Chemistry* in *International Conference on Machine Learning* **70** (PMLR, 2017), 1263–1272.
3. Xie, T. & Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical Review Letters* **120,** 145301 (2018).
4. Deng, B., Zhong, P., Jun, K., Riebesell, J., Han, K., Bartel, C. J. & Ceder, G. CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence* **5,** 1031–1041 (2023).
5. Yu, H., Zhong, Y., Hong, L., Xu, C., Ren, W., Gong, X. & Xiang, H. Spin-dependent graph neural network potential for magnetic materials. *Physical Review B* **109,** 144426 (2024).
6. Satorras, V. G., Hoogeboom, E. & Welling, M. *E(n) Equivariant Graph Neural Networks* in *International Conference on Machine Learning* **139** (PMLR, 2021), 9323–9332.
7. Liu, X., Zhang, J. & Pei, Z. Machine learning for high-entropy alloys: Progress, challenges and opportunities. *Progress in Materials Science* **131,** 101018 (2023).
8. Taheri, M. L. *et al.* Understanding and leveraging short-range order in compositionally complex alloys. *MRS Bulletin* **48,** 1280–1291 (2023).
9. Rao, Z. *et al.* Machine learning–enabled high-entropy alloy discovery. *Science* **378,** 78–85 (2022).

10. Liu, D. *et al.* Exceptional fracture toughness of CrCoNi-based medium- and high-entropy alloys at 20 kelvin. *Science* **378,** 978–983 (2022).
11. Batchelor, T. A. A., Pedersen, J. K., Winther, S. H., Castelli, I. E., Jacobsen, K. W. & Rossmeisl, J. High-Entropy Alloys as a Discovery Platform for Electrocatalysis. *Joule* **3,** 834–845 (2019).
12. Lu, Z., Chen, Z. W. & Singh, C. V. Neural Network-Assisted Development of High-Entropy Alloy Catalysts: Decoupling Ligand and Coordination Effects. *Matter* **3,** 1318–1333 (2020).
13. Chen, Z. W., Gariepy, Z., Chen, L., Yao, X., Anand, A., Liu, S.-J., Tetsassi Feugmo, C. G., Tamblyn, I. & Singh, C. V. Machine-learning-driven high-entropy alloy catalyst discovery to circumvent the scaling relation for CO_2 reduction reaction. *ACS Catalysis* **12,** 14864–14871 (2022).
14. Vazquez, G., Singh, P., Sauceda, D., Couperthwaite, R., Britt, N., Youssef, K., Johnson, D. D. & Arróyave, R. Efficient machine-learning model for fast assessment of elastic properties of high-entropy alloys. *Acta Materialia* **232,** 117924 (2022).
15. Wang, X., Tran, N.-D., Zeng, S., Hou, C., Chen, Y. & Ni, J. Element-wise representations with ECNet for material property prediction and applications in high-entropy alloys. *npj Computational Materials* **8,** 253 (2022).
16. Zhang, H., Lai, T., Chen, J., Manthiram, A., Rondinelli, J. M. & Chen, W. Learning molecular mixture property using chemistry-aware graph neural network. *PRX Energy* **3,** 023006 (2024).
17. Heid, E., Greenman, K. P., Chung, Y., Li, S.-C., Graff, D. E., Vermeire, F. H., Wu, H., Green, W. H. & McGill, C. J. Chemprop: A Machine Learning Package for Chemical Property Prediction. *Journal of Chemical Information and Modeling* **64,** 9–17 (2024).
18. Choudhary, K. & Decost, B. Atomistic Line Graph Neural Network for improved materials property predictions. *npj Computational Materials* **7,** 185 (2021).
19. Chen, C. & Ong, S. P. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science* **2,** 718–728 (2022).
20. Batatia, I., Kovacs, D. P., Simm, G., Ortner, C. & Csanyi, G. *MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields* in *Neural Information Processing Systems* **35** (Curran Associates, Inc., 2022), 11423–11436.
21. Lin, Y., Yan, K., Luo, Y., Liu, Y., Qian, X. & Ji, S. *Efficient Approximations of Complete Interatomic Potentials for Crystal Property Prediction* in *International Conference on Machine Learning* **202** (PMLR, 2023), 21260–21287.
22. Schütt, K. T., Unke, O. T. & Gastegger, M. *Equivariant message passing for the prediction of tensorial properties and molecular spectra* in *International Conference on Machine Learning* **139** (PMLR, 2021), 9377–9388.
23. Yan, K., Liu, Y., Lin, Y. & Ji, S. *Periodic Graph Transformers for Crystal Material Property Prediction* in *Neural Information Processing Systems* **35** (Curran Associates, Inc., 2022), 15066–15080.
24. Gao, T., Song, H., Wang, B., Gao, Y., Liu, Y., Xie, Q., Chen, Q., Xiao, Q. & Liang, Y. Molecular dynamics simulations of tensile response for FeNiCrCoCu high-entropy alloy with voids. *International Journal of Mechanical Sciences* **237,** 107800 (2023).
25. Yuan, G., Wu, M. & Ruiz Pestana, L. Density functional theory-machine learning characterization of the adsorption energy of oxygen intermediates on high-entropy alloys made of earth-abundant metals. *The Journal of Physical Chemistry C* **127,** 15809–15818 (2023).
26. Widom, M. Modeling the structure and thermodynamics of high-entropy alloys. *Journal of Materials Research* **33,** 2881–2898 (2018).
27. Zhao, S., Jiang, B., Song, K., Liu, X., Wang, W., Si, D., Zhang, J., Chen, X., Zhou, C., Liu, P., *et al.* Machine learning assisted design of high-entropy alloys with ultra-high microhardness and unexpected low density. *Materials & Design* **238,** 112634 (2024).
28. Araujo, R. B. & Edvinsson, T. Supervised AI and Deep Neural Networks to Evaluate High-Entropy Alloys as Reduction Catalysts in Aqueous Environments. *ACS Catalysis* **14,** 3742–3755 (2024).
29. Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R. & Smola, A. J. *Deep Sets* in *Neural Information Processing Systems* **30** (Curran Associates, Inc., 2017).
30. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. *Attention Is All You Need* in *Neural Information Processing Systems* **30** (Curran Associates, Inc., 2017).

31. Zhang, J., Cai, C., Kim, G., Wang, Y. & Chen, W. Composition design of high-entropy alloys with deep sets learning. *npj Computational Materials* **8,** 89 (2022).
32. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., *et al.* Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12,** 2825–2830 (2011).
33. Loshchilov, I. & Hutter, F. *Decoupled Weight Decay Regularization* in *International Conference on Learning Representations* (2019).
34. Meyers, M. A. & Chawla, K. K. in *Mechanical Behavior of Materials* 71–160 (Cambridge University Press, 2008).
35. Li, K., Persaud, D., Choudhary, K., Decost, B., Greenwood, M. & Hattrick-Simpers, J. Exploiting redundancy in large materials datasets for efficient machine learning with less data. *Nature Communications* **14,** 7283 (2023).
36. Shang, Y., Brechtl, J., Pistidda, C. & Liaw, P. K. in *High-Entropy Materials: Theory, Experiments, and Applications* (eds Brechtl, J. & Liaw, P. K.) 435–522 (Springer International Publishing, Cham, 2021).
37. An, Z. *et al.* Negative mixing enthalpy solid solutions deliver high strength and ductility. *Nature* **625,** 697–702 (2024).
38. Harada, J. K., Charles, N., Poeppelmeier, K. R. & Rondinelli, J. M. Heteroanionic materials by design: progress toward targeted properties. *Advanced Materials* **31,** 1805295 (2019).
39. Saber, M., Reynolds, C., Li, J., Pollock, T. M. & Van der Ven, A. Chemical and structural factors affecting the stability of Wadsley–Roth block phases. *Inorganic Chemistry* **62,** 17317–17332 (2023).
40. Rondinelli, J. M. & May, S. J. Deliberate deficiencies: Expanding electronic function through non-stoichiometry. *Matter* **1,** 33–35 (2019).
41. Kim, L., Scougale, W. R., Sharma, P., Shirato, N., Wieghold, S., Rose, V., Chen, W., Balasubramanian, G. & Chien, T. Distinguishing Elements at the Sub-Nanometer Scale on the Surface of a High Entropy Alloy. *Advanced Materials* **36,** 2402442 (2024).
42. Zhang, W., Chabok, A., Kooi, B. J. & Pei, Y. Additive manufactured high entropy alloys: A review of the microstructure and properties. *Materials & Design* **220,** 110875 (2022).
43. Biewald, L. *Experiment Tracking with Weights and Biases* Software available from wandb.com. 2020.
44. Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G. & Grohe, M. *Weisfeiler and Leman go neural: Higher-order graph neural networks* in *AAAI Conference on Artificial Intelligence* **33** (2019), 4602–4609.

A Algorithm and computing details

Algorithm 1: LESets machine learning model.

Input: $X = \{(x_i, w_i)\}$; LE graphs $x_i = (V, E)$, with nodes $V = \{v_m\}$ and edges $E = \{e_{mn}\}$
Output: Property prediction $y = f(X)$

(1) ϕ module: Message-passing graph neural network.
for $x_i \in X$ **do**
  for $l = 1, \ldots,$ `n_conv_layers` **do**
    $v_m \leftarrow$ `message_passing`(v_m, v_n, e_{mn}) for $v_m \in V$ and $n \in \text{neighbors}(m)$
    $v_m \leftarrow$ `activation`(v_m)
  end
  Readout: $z_i \leftarrow$ `mean_pooling`$(v_m \in V)$
  FC layer: $z_i \leftarrow \tanh(W z_i + b)$ `/* individual representations */`
end

(2) $\oplus$ module: Weighted sum and (optionally) attention mechanism.
for all z_i **do**
  if `use_att` **then**
    $q \leftarrow W^q z_i, k \leftarrow W^k z_i, v \leftarrow W^v z_i$
    $z_i \leftarrow$ `softmax` $\left(q \cdot k^{\mathrm{T}} / \sqrt{d_k}\right) \cdot v$
  end
end
$Z \leftarrow \sum_i w_i z_i$ `/* aggregated representation */`

(3) ρ module: L-layer fully connected (FC) neural network.
for $l = 1, \ldots, L-1$ **do**
  $Z \leftarrow$ `activation`$(W^l Z + b^l)$
end
$y \leftarrow W^L Z + b^L$
return y `/* value predicted */`

Computing details. Experiments are performed on a Linux workstation with Intel Xeon W-2295 CPU (18 cores, 3.0 GHz), NVIDIA Quadro RTX 5000 GPU (16 GB memory), and 256 GB of RAM. Training of LESets typically takes 10–20 minutes on one GPU, with variations depending on data size, hyperparameters, and epochs before termination.

B Supplemental results

Figure 7 shows a typical learning curve of LESets.

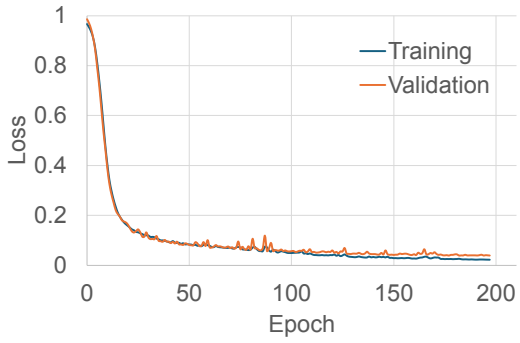

Figure 7: A typical learning curve of the LESets model.

Hyperparameter tuning and experiment tracking are performed using the weights and biases platform [43]; the tuning results are listed in Table 1.

Hyperparameter	Young's	Bulk	r_{ws}
Convolution operator	GraphConv [44]	CGConv [3]	CGConv
Convolution layers	2	3	2
FC layers	3	3	3
Hidden dimension	32	32	32
Use attention	True	False	False

Table 1: Tuned hyperparameters of LESets for each target property.

Table 2 lists the complete results of model comparisons.

Model	Young's (GPa)		Bulk (GPa)		r_{ws} (Å)	
	MAE	R^2	MAE	R^2	MAE	R^2
LESets-att	**8.891**	**0.920**	7.922	0.821	0.011	0.989
LESets-ws	10.03	0.902	**7.685**	**0.828**	**0.010**	**0.989**
Deep Sets [31]	11.19	0.884	7.984	0.819	0.011	0.989
gradient boost	11.18	0.878	9.935	0.764	0.015	0.978
random forest	14.91	0.813	10.87	0.725	0.019	0.966
Lasso	13.65	0.838	10.75	0.742	0.022	0.957
ridge regression	13.62	0.839	10.75	0.741	0.021	0.959
kNN	15.89	0.793	11.66	0.728	0.022	0.951
SVM	23.14	0.556	12.01	0.662	0.013	0.984

Table 2: Average MAE and R^2 across 30 replicates of all tested models on Young's modulus, bulk modulus, and Wigner-Seitz radius r_{ws}. The best performance metric on each property is indicated by boldface.