Computational Materials Science 158 (2019) 414–419

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Machine learning of ab-initio energy landscapes for crystal structure predictions

Shreyas Honrao[a,c,1], Bryan E. Anthonio[b,1], Rohit Ramanathan[a], Joshua J. Gabriel[c], Richard G. Hennig[c,a,*]

[a] Department of Materials Science and Engineering, Cornell University, Ithaca, NY 14850, USA
[b] Department of Applied and Engineering Physics, Cornell University, Ithaca, NY 14850, USA
[c] Department of Materials Science and Engineering, University of Florida, Gainesville, FL 32611, USA

ARTICLE INFO

Keywords:
Machine learning
Formation energies
Kernel ridge regression
Support vector regression
Partial radial distribution functions
Crystal structure predictions
Genetic algorithm
Binary systems
Li-Ge compounds
Density-functional theory

ABSTRACT

We present a machine learning approach to calculate unrelaxed and relaxed formation energies of compounds relative to the ground state crystal structure of the pure components in the context of structure predictions in binary systems. Typical methods for structure predictions such as genetic algorithms often rely on density-functional theory codes to perform such calculations at a relatively high computational cost. In this work, we explore two commonly used kernel-based learning algorithms, kernel ridge regression and support vector regression. The efficiency of machine learning approaches relies on suitable data representations that encode the relevant physical information about the crystal structures. We select partial radial distribution functions to represent this structural information. We apply the machine learning approaches to the binary Li-Ge system and show that these methods provide small root-mean square prediction errors of about 20 meV/atom across the composition and structure space. Furthermore, we demonstrate that the model can be trained to predict the formation energies of the relaxed structures with the same accuracy when given unrelaxed structures as input. The high accuracy for the prediction of the relaxed energies of unrelaxed structures suggests that the machine-learning method can eliminate unlikely candidate structures from a genetic algorithm search, thus reducing the computational cost required for the explorations of energy landscapes and improving the performance of genetic algorithms for structure predictions.

1. Introduction

Predicting the crystal structure of a material given just its constituent elements is a crucial first step in computational materials design, as a material's structure dictates many of its macroscopic properties. The problem of predicting the structure of a material amounts to finding the configuration of atoms that minimizes the total energy of the structure. Using global search algorithms like the genetic algorithm (GA), the entire phase diagram of a system can be computed [1,2]. However, the bottleneck for predicting the crystal structures across the phase diagram for a given system lies in the computational expense of the required total energy computations using density-functional theory (DFT).

Machine learning (ML) models offer the potential to accelerate the structure search by quickly identifying low-energy candidate structures, which can then be evaluated by more accurate and computationally intensive DFT calculations [3–6]. ML makes use of existing data to create predictive models that generate reasonable output values for unseen inputs. Such models have recently been employed to calculate atomization energies of organic molecules [7–9], predict electronic properties [3,10], and model electronic quantum transport [11], among other things [12–17].

The two most important aspects of any ML model are the data representation and the learning algorithm. An appropriate representation of input data is critical to the generation of an accurate model. In this work, we are interested in learning unrelaxed and relaxed crystal structure formation energies relative to the pure components across the entire compositional and configurational space of binary systems. Our problem is unique in the way that we have multiple structures of the same composition whose formation energy we are trying to predict. These formation energies depend non-linearly on the coordination number and types of bonds for each atom and cannot be approximated

* Corresponding author at: Department of Materials Science and Engineering, University of Florida, Gainesville, FL 32611, USA.
E-mail address: rhennig@ufl.edu (R.G. Hennig).
[1] These authors contributed equally.

https://doi.org/10.1016/j.commatsci.2018.08.041
Received 29 May 2018; Received in revised form 21 August 2018; Accepted 22 August 2018
Available online 28 November 2018
0927-0256/

using simple representations that rely on counting the number of different bonds or units [3,18]. Global or macro descriptors that use physical and chemical properties like atomic number, band gap, electronegativity, oxidation state, elastic moduli, etc. [19,20] are also not suitable as they fail to capture the local structural information. In this work, we show that the use of partial radial distribution functions (RDF) enables the accurate prediction of the unrelaxed and relaxed formation energies of crystal structures spanning the entire phase diagram of the binary Li-Ge system. We compare the relative efficiency of two commonly used learning algorithms, kernel ridge regression (KRR) and support vector regression (SVR). Both algorithms predict the energy of the Li-Ge phases to within 20 meV/atom. We find that the SVR algorithm scales better for large datasets compared to KRR, making it more preferable for quickly predicting formation energies, thereby minimizing the number of DFT calculations required for explorations of energy landscapes to identify low-energy minima.

2. Dataset

We use a dataset of approximately 14,000 Li-Ge structures generated using our Genetic Algorithm for Structure Prediction (GASP) [2,21,22] in our search for novel compounds in the Li-Ge system, a promising battery material [23]. GASP starts with an initial generation of structures and uses DFT to evaluate their energies. Low-energy structures are preferentially used to generate a subsequent generation of structures, which are then evaluated using DFT. This process repeats until the energies of the structures converge to some minimum value or a predefined number of structure evaluations is reached. Relevant details can be found in Ref. [23]. We expect the data from GA searches with GASP to be significantly more diverse than data from molecular dynamics (MD) simulations as the GA samples the overall global energy landscape as opposed to just exploring nearby local minima of the energy landscape. This enhanced sampling diversity reduces the amount of correlation between structures in our GASP dataset compared to MD datasets.

The relevant energy for structure predictions as a function of composition is the formation energy relative to the crystal structure of the pure components,

$$E_f = E_{tot} - X_{Li}E_{Li} - X_{Ge}E_{Ge}, \quad (1)$$

where E_{tot} is the total energy per atom of the Li-Ge crystal structure, X_{Li} and X_{Ge} are the molar fractions of Li and Ge in the structure, and E_{Li} and E_{Ge} the energies per atom of pure Li and Ge. We use the term 'unrelaxed' formation energy (E_f^u) while referring to the energy of the unrelaxed structure, whereas 'relaxed' formation energy (E_f^r) refers to the energy of the minimum-energy structure obtained upon relaxation. For the application of the ML approach, we select structures from the GASP search with unrelaxed formation energies $E_f^u < 200$ meV/atom, resulting in a total of 14,168 crystal structures.

To ensure that the structures from a particular GASP relaxation run are either all in the training or testing set, we group the structures according to their basin of attraction, *i.e.,* the minimum-energy structure they relax to. This splitting of the data further reduces the correlation between structures in the testing and training set and provides a more stringent and realistic test of the ML methods. We refer to the groups as basin groups. The 14,168 crystal structures in the dataset are thus split into 679 basin groups.

The Li-Ge dataset is available online at https://materialsweb.org [24], our database of structural, electronic and thermodynamic data for 2D/3D materials, powered by MPInterfaces [25] and pymatgen [26].

3. Data representation

ML regression models take a vector $x \in \mathbb{R}^n$ as input and return a value y. To utilize these models for energy predictions, we must first construct a vector-based data representation of the crystal structures that encodes the relevant physical information, *i.e.,* the chemical identity and position of the atoms. This data representation should ideally fulfill three criteria: (i) *invariance* with respect to the choice of unit cell and crystal symmetry, (ii) *uniqueness*, so no two different crystal structures have the same vector representation, and (iii) *continuity*, such that the energy difference between two crystal structures with vector representations x_1 and x_2 goes to zero in the limit $||x_1 - x_2|| \to 0$. A number of different data representations have been studied by materials researchers over the last few years - including symmetry functions [27], smooth overlap of atomic positions [28], Fourier transform of radial basis functions [9], and other ad hoc descriptors - that fulfill one or more of these criteria [7,10,29–33]. Here, we explore a representation using partial RDFs that satisfies invariance and continuity but is not necessarily unique.

The partial RDF, $g_{AB}(r)$, captures the average distribution of interatomic distances $d_{ij}^{AB} = |\vec{r}_i^{\,A} - \vec{r}_j^{\,B}|$ between atoms i and j of type A and B:

$$g_{AB}(r) = \frac{1}{N_A}\sum_{i=1}^{N_A}\sum_{j=1}^{\infty}\frac{1}{r^p}\exp\left[-\frac{(r - d_{ij}^{AB})^2}{2\sigma_g^2}\right]\Theta(d_c - d_{ij}^{AB}). \quad (2)$$

The first sum includes all N_A atoms of type A within the unit cell. The second sum is over all atoms of type B up to a cutoff distance d_c, which is enforced by the Heaviside function, $\Theta(d_c - d_{ij}^{AB})$. The cutoff distance, d_c, is chosen such that it extends beyond the unit cell of the crystal structure to ensure that the data representation captures the periodicity of the crystal structure. The $1/r^p$ term renormalizes the partial RDFs as a function of distance such that for $p = 2$ it approaches a constant at large distances and for $p > 2$ it decays with distance. The width of the Gaussian, σ_g, used for broadening the distribution, is a free parameter, which we set to $\sigma_g = 0.2$ Å.

Binning transforms the partial RDFs into a discrete representation. We select the bin width equal to, or smaller than, the width of the Gaussian broadening of the RDF, σ_g, to mitigate the loss of information incurred in the binning process. The value, $\hat{g}_{AB}^{j}$, of each bin j is taken as the function $g_{AB}(r)$ evaluated at the right edge of the respective bin. The final input representation for the machine learning model is the vector $X = (\hat{g}_{AB}^{j})\ \forall\ (j, A, B)$. Fig. 1 visualizes an example structure, Li_4Ge, and its corresponding partial RDFs.

4. Algorithms

We explore two commonly used kernel-based learning methods for the prediction of the formation energies: (i) kernel ridge regression (KRR) and (ii) support vector regression (SVR). We use the

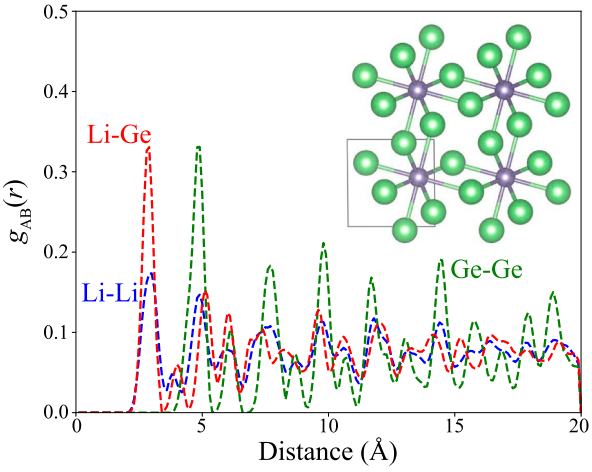

Fig. 1. The Li_4Ge crystal structure and the corresponding partial radial distribution functions $g_{AB}(r)$ with $d_c = 20$ Å and $p = 2$.

implementations of the KRR and SVR algorithms available in the sci-kit learn library for Python [34]. Given a dataset of N structures $\{(x_1, y_1), (x_2, y_2)...(x_N, y_N) \mid x_i \in \mathbb{R}^n, y_i \in \mathbb{R}\}$, where x_i are the independent and y_i the dependent variables, both algorithms operate by first implicitly mapping the input vectors $x_i \in \mathbb{R}^n$ to a higher dimensional space $\phi(x_i) \in \mathbb{R}^m$ (where $m > n$), also known as the function space. Instead of fitting the linear model in the original space, we fit this model to the data presented in the function space. The linear model fitted in function space is nonlinear in the original space and more accurately captures the non-linearities of the input data [35].

The kernel trick allows for the evaluation of inner products in this higher dimensional space without computing the mapping explicitly by using a kernel function $k(x, x')$, such that $\langle \phi(x), \phi(x') \rangle = k(x, x')$. A common choice for the kernel function and the one we use in our regression model is the Gaussian radial basis kernel,

$$k(x, x') = \exp\left(-\frac{|x - x'|^2}{2\sigma_k^2}\right). \tag{3}$$

Here, $|x - x'|$ is the Frobenius norm of the difference between the input vector variables x and x', and σ_k is the kernel width which must be optimized when fitting the model to the input training data.

For the ML of the energy landscape of crystal structures, the input vectors x_i are the partial RDF vectors X_i for each structure i. The dependent variables y_i are the energies of the structure. The kernel function $k(X, X')$ provides a measure of similarity between two structures in configuration space, such that $k(X, X') \to 1$ as the structures become increasingly more similar.

4.1. Kernel ridge regression

The simplest regression algorithm that employs a kernel is KRR, a nonlinear variation of ridge regression. The algorithm yields the function

$$E^{\text{est}}(x) = \sum_{i=1}^{N} \alpha_i k(x, x_i), \tag{4}$$

such that $E^{\text{est}}(x_i) \approx y_i$. The function E^{est} takes the representation of a crystal structure as input and yields a prediction of the structure's formation energy. The parameter vector $\alpha \in \mathbb{R}^N$ is obtained by solving the optimization problem

$$\min_{\alpha} \sum_{i=1}^{N} [E^{\text{est}}(x_i) - y_i]^2 + \lambda \sum_{i=1}^{N} \alpha_i^2, \tag{5}$$

where $\lambda \in \mathbb{R}$ is the regularization parameter that penalizes the norm of α to prevent overfitting of the training data. Like the kernel width, σ_k, the regularization parameter, λ, is optimized during the learning process.

One drawback to the KRR algorithm is that it does not scale well to large datasets since the dimension of α increases with the size, N, of the input training data.

4.2. Support vector regression

SVR presents an alternative to KRR, that has the advantage of fast prediction even for large datasets. Our discussion and notation of SVR follow Ref. [36]. The ϵ-SVR algorithm attempts to find a function $f(x) = \langle w, x \rangle + b$ that deviates at most ϵ from the training data, y_i, and generalizes well. We introduce slack variables ξ and ξ^* to allow for some errors and obtain the following optimization problem:

$$\text{minimize } \tfrac{1}{2}|w|^2 + C \sum_{i=1}^{N} (\xi_i + \xi_i^*)$$
$$\text{subject to } \begin{cases} y_i - [\langle w, x_i \rangle + b] \leqslant \epsilon + \xi_i \\ [\langle w, x_i \rangle + b] - y_i \leqslant \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geqslant 0. \end{cases} \tag{6}$$

The parameter C is a regularization parameter that determines the extent to which deviations greater than ϵ are tolerated.

An interesting feature of the ϵ-SVR algorithm is that in solving the quadratic problem, several terms go to zero, resulting in the solution being sparse. This is unlike the KRR algorithm, where the solution depends on the entire set of training data. As a result, the prediction stage of the SVR algorithm is considerably faster, particularly for large datasets.

5. Model selection and validation

5.1. Input data pre-processing

Before training the ML models, we perform feature scaling on the components of input vectors in the training set $\{(x_1, y_1), (x_2, y_2)...(x_N, y_N) \mid x_i \in \mathbb{R}^n, y_i \in \mathbb{R}\}$ by subtracting the means and dividing by the standard deviations:

$$x'_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}, \tag{7}$$

$$\text{where } \sigma_j = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_{ij} - \mu_j)^2} \text{ and } \mu_j = \frac{1}{N} \sum_{i=1}^{N} x_{ij}.$$

This transformation ensures that each set of components $\{x_{1j}, x_{2j}, ..., x_{Nj}\}$ has zero mean and unit variance. Standardizing each set of components this way is necessary to avoid the Frobenius norm from being biased towards vector components with higher variance.

Next, we partition the dataset of basin groups into a training set and a testing set, through random assignment. Of the 679 basin groups, we set aside 200 for training, and the remaining 479 go into the testing set. The structures in the training set are used for training the ML model and learning the optimal parameters and hyperparameters, whereas we use the structures from the testing set to characterize the predictive accuracy of the learned models. It is important to note that structures from the same basin group may be quite similar to each other. In order to avoid very similar structures from the same group ending up in different sets, we partition our dataset by basin groups rather than individual structures. Since the number of structures in each basin group is not necessarily the same, different partitions result in slightly different sizes of training and testing sets.

5.2. Cross-validation

Determining the coefficients for SVR and KRR first requires appropriate selection of the hyperparameters C and λ, as well as the kernel width σ_k. This is achieved through the process of cross-validation. Hyperparameters for both the SVR and KRR algorithms are selected using 10-fold cross-validation on the training set of 200 randomly

Table 1
Comparison of average errors (MAE and RMSE) in meV/atom and goodness of fit (R^2) for the SVR and KRR algorithms.

Algorithm	MAE	RMSE	R^2value
KRR-unrelaxed	12.7	20.4	0.981
KRR-relaxed	11.8	20.3	0.980
SVR-unrelaxed	13.6	20.8	0.979
SVR-relaxed	13.4	20.9	0.979

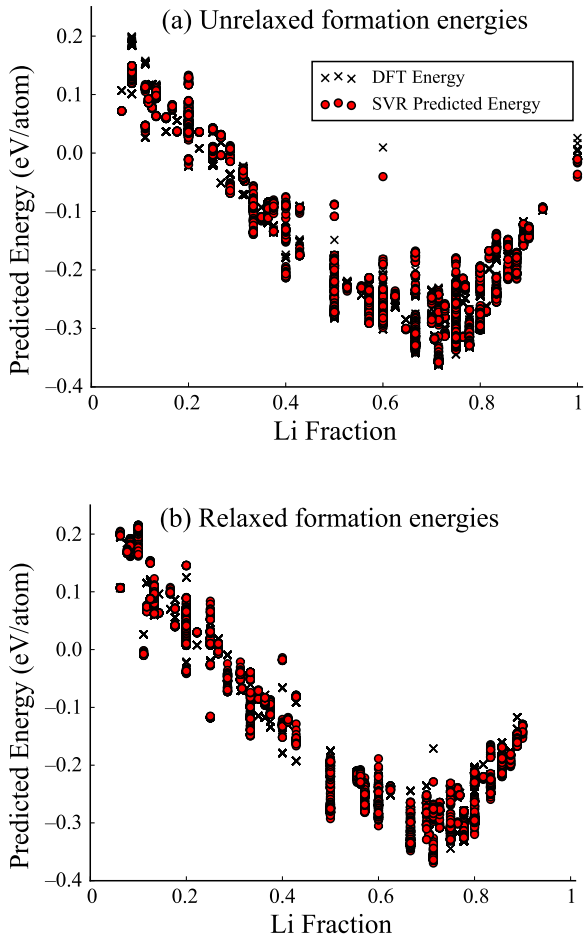

Fig. 2. Phase diagrams of the Li-Ge system showing (a) unrelaxed and (b) relaxed formation energies obtained from SVR vs. DFT-calculated formation energies of structures in the testing set.

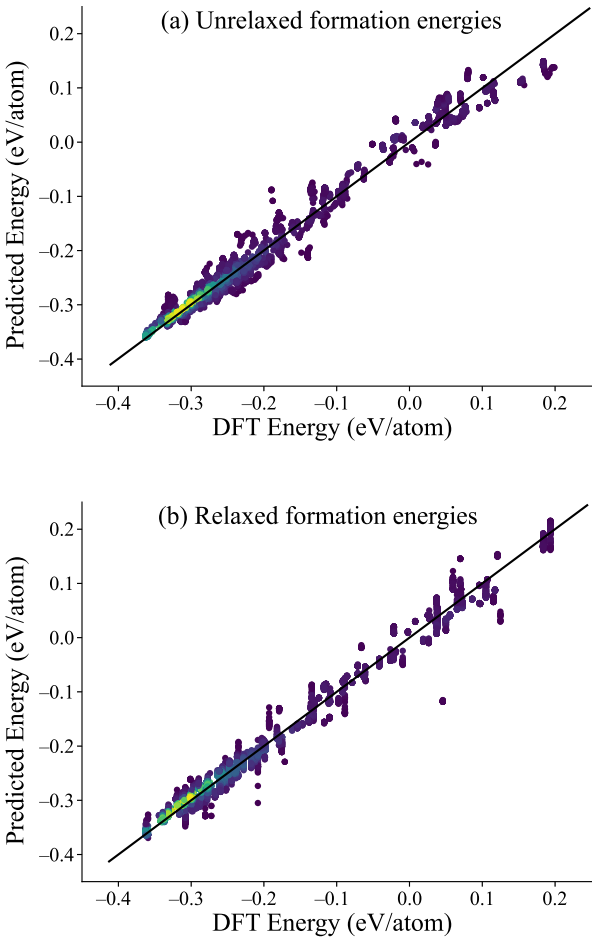

Fig. 3. SVR-predicted vs. DFT-calculated (a) unrelaxed and (b) relaxed formation energies of structures in the testing set.

assigned basin groups. For each of the 10 folds, the hyperparameters are randomly sampled from exponential distributions ($P(x) = \beta e^{-\beta x}$) and used to train the models. A random search of parameters has been shown to be more efficient than a grid search for hyperparameter optimization [37].

For KRR, λ is sampled from an exponential distribution with $\beta = 1$, while σ_k is sampled from an exponential distribution with $\beta = 0.05$. For SVR, C is sampled from an exponential distribution with $\beta = 0.2$. For the hyperparameter σ_k, the SVR implementation in the scikit-learn module samples a parameter $\gamma = 1/2\sigma_k^2$ from an exponential distribution scaled with $\beta = 1000$. Once the model hyperparameters are selected using cross-validation, they are used to train the KKR and SVR models on the structures from the training set. The trained models are then used to predict the formation energies of the remaining structures in the testing set.

6. Parameter selection

6.1. Data representation

The structure representation of Eq. (2) contains besides σ_g, which we have already set to 0.2 Å, two other parameters that we control: (i) the cutoff distance for the Heaviside step function, d_c, and (ii) the exponent of the $1/r^p$ term, p. We train the SVR and KRR models with $d_c = 5,\ 10,\ 15$, and 20 Å, and find the root mean squared error (RMSE) to be similar in all cases, except for the lowest $d_c = 5$ Å, where the error was higher. This indicates that $d_c = 10$ Å is an optimal choice for our model. We also train ML models with different values of $p = 0,\ 1,\ 2,\ 3,\ 4$. In the absence of feature scaling as shown in Eq. (7), we observe that the normalization of the partial RDFs by $1/r^2$ would result in the lowest error. However, feature scaling completely offsets the effect of the $1/r^p$ term and ensures that the errors are independent of the choice of p. For all further work, we use a default value of $p = 2$.

6.2. ϵ–SVR: choice of ϵ

The value of ϵ defines a margin of tolerance around the actual target values of points in the training set. We test different values of ϵ from 1 to 40 meV/atom within our SVR model to determine the optimal value. As expected, we find that lower values of ϵ generally result in a lower RMSE. For $\epsilon < 10$ meV/atom, the change becomes negligible. Since smaller values of ϵ also make the SVR model more complex and prone to overfitting, we use an error tolerance of $\epsilon = 10$ meV/atom that gives us the best RMSE vs. complexity tradeoff.

7. Results

Using the procedures and parameters determined above, we optimize SVR and KRR models on our training set to predict unrelaxed formation energies of crystal structures in the testing set. Going a step further, we are also successful at training our models to predict relaxed formation energies - formation energies of the minimum-energy configurations, or basins, that input structures relax to. This makes it possible to predict whether a new structure generated by a GA will relax to a low-energy configuration or not. Without this ability, the ML energy model would be counterproductive for a GA-based approach to

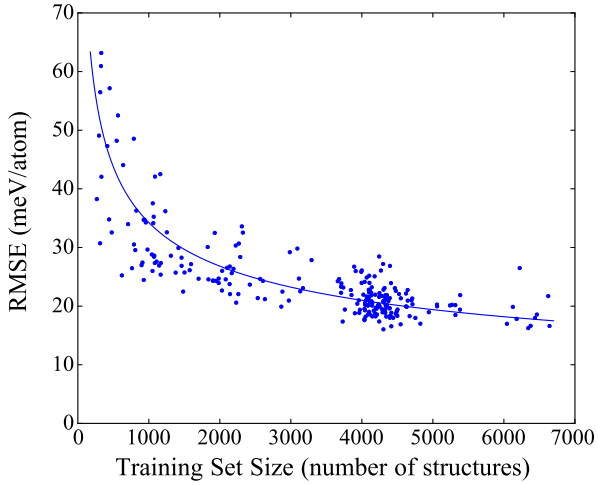

Fig. 4. Learning curve showing the prediction error on the testing set (RMSE) vs. the size of the training set (number of structures) for predicting unrelaxed formation energies using the SVR algorithm.

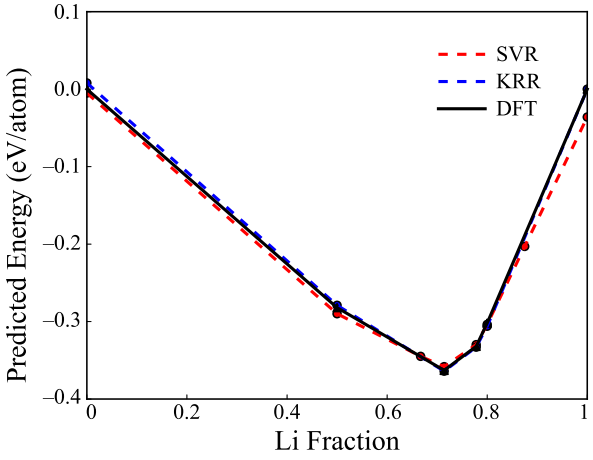

Fig. 5. Comparison of the convex hulls constructed using only machine-learned formation energies (from both the training and testing sets) with the actual DFT-calculated convex hull for Li-Ge.

structure prediction.

We calculate the average formation energy prediction errors by performing trials over 100 different partitions of training and testing sets. Table 1 shows the average RMSE, mean absolute error (MAE), and R^2 value of the fit for the two ML algorithms. The average RMSE for predicting unrelaxed energies is 20.4 meV/atom for KRR and 20.8 meV/atom for SVR. For predicting relaxed formation energies, it is 20.3 meV/atom for KRR and 20.9 meV/atom for SVR. This is significant because the errors are within 1 kcal/mol or 43 meV/atom required for chemical accuracy [30].

Fig. 2 compares the predicted unrelaxed and relaxed formation energies in the Li-Ge system with the DFT data for the SVR algorithm. Note that only structures from the testing set are shown in these figures. The lack of any clear outliers in either phase diagram indicates that the partial RDF structure representation provides good estimates of the unrelaxed and relaxed formation energies of structures across the entire composition space.

Fig. 3 shows the overall predictive accuracy of the SVR algorithms. The tight clustering of points about the main diagonal indicates that the ML model reasonably predicts the formation energies given by DFT. Based on the accuracy of our surrogate ML models in Table 1 and the results from Fig. 3 we can already infer that it might be possible to eliminate unlikely structures from a GA run based solely on predictions made by the ML models.

Fig. 4 shows how the RMSE for predicting unrelaxed formation energies using the SVR model depends on the training set size. As expected, the average prediction error decreases with increasing training set size. For training sets with a few thousand structures, the error quickly converges to 20 meV/atom. Interestingly, even for smaller training set sizes of 500 to 1000 structures, the errors range from 30 to 50 meV/atoms, which could already be useful for the screening of structures in a GA search. Our choice of the training set size of 200 basin groups, which corresponds to about 4000–4500 structures, simultaneously optimizes the speed and accuracy of our ML models.

Just like the average KRR errors in Table 1, we observe that other results from the KRR model also look very similar to the corresponding SVR results in Figs. 2–4. We show figures only from the SVR model here for the sake of brevity.

Fig. 5 compares the ML-predicted convex hulls for both models to the actual convex hull constructed from DFT calculations. While our goal is to use ML models to simply minimize the number of DFT calculations, leaving the more accurate final energy calculations to DFT, it is encouraging to see that both ML models do a fair job of predicting the stable phases in the Li-Ge system. It should be noted that structures from both sets, training and testing, are used to construct the convex hull.

8. Summary

In summary, we presented a machine learning approach for predicting unrelaxed and relaxed formation energies relative to the ground state crystal structure of the pure components in the context of structure predictions in binary systems. The basic concept is to use a dataset of known crystal structures whose energies have been evaluated by DFT to generate a new surrogate energy model capable of accurately predicting formation energies of unseen structures outside of this dataset. We illustrated a new means of representing crystal structures for machine learning applications using partial radial distribution functions and demonstrated two machine learning models capable of yielding energy predictions within chemical accuracy of 1 kcal/mol using a dataset consisting of 14,000 Li-Ge crystal structures. Our accuracy suggests that we might be able to eliminate unlikely structures from a GA run based solely on predictions made by the ML model. Our work is one of the first attempts at learning the entire energy landscape of a binary material with the use of machine learning. Future work will involve coupling a genetic algorithm with our machine learning surrogate energy model to more efficiently explore energy landscapes and identify low-energy minima.

9. Data availability

The data required to reproduce these findings are available to download from https://materialsweb.org.

10. CRediT authorship contribution statement

Shreyas Honrao: Conceptualization, Methodology, Software, Validation, Formal Analysis, Data Curation, Writing, Visualization. **Bryan E. Anthonio**: Conceptualization, Methodology, Software, Validation, Writing, Visualization. **Rohit Ramanathan**: Methodology, Software, Validation. **Joshua J. Gabriel**: Software, Data Curation. **Richard G. Hennig**: Conceptualization, Methodology, Validation, Writing, Visualization, Supervision, Project Administration, Funding Acquisition.

Acknowledgments

This work was supported by the National Science Foundation under Grant Nos. DMR-1542776 and ACI-1440547. BEA was supported by the

Cornell University Louis Stokes Alliances for Minority Participation, a NSF supported program under Grant No. 1202480 and a grant provided by the Semiconductor Research Corporation with Support from Intel Foundation through Cornell Engineering Learning Initiatives. This research used computational resources provided by the University of Florida Research Computing (http://researchcomputing.ufl.edu) and the Texas Advanced Computing Center under Contracts TG-DMR050028N, TG-DMR140143, and TG-DMR150006. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation Grant No. ACI-1053575.

References

[1] G. Trimarchi, A.J. Freeman, A. Zunger, Predicting stable stoichiometries of compounds via evolutionary global space-group optimization, Phys. Rev. B 80 (2009) 092101.
[2] W.W. Tipton, R.G. Hennig, A grand canonical genetic algorithm for the prediction of multi-component phase diagrams and testing of empirical potentials, J. Phys. Condens. Matter 25 (49) (2013) 495401.
[3] G. Pilania, C. Wang, X. Jiang, S. Rajasekaran, R. Ramprasad, Accelerating materials property predictions using machine learning, Sci. Rep. 3 (2013) 2810.
[4] M. Rupp, Machine learning for quantum mechanics in a nutshell, Int. J. Quantum Chem. 115 (16) (2015) 1058–1073.
[5] V. Botu, R. Ramprasad, Adaptive machine learning framework to accelerate ab initio molecular dynamics, Int. J. Quantum Chem. 115 (16) (2015) 1074–1083.
[6] D. Xue, P.V. Balachandran, J. Hogden, J. Theiler, D. Xue, T. Lookman, Accelerated search for materials with targeted properties by adaptive design, Nat. Commun. 7 (2016) 11241.
[7] M. Rupp, A. Tkatchenko, K.-R. Müller, O.A. von Lilienfeld, Fast and accurate modeling of molecular atomization energies with machine learning, Phys. Rev. Lett. 108 (2012) 058301.
[8] H. Huo, M. Rupp, Unified representation for machine learning of molecules and crystals, available from: arXiv preprint <arXiv:1704.06439>.
[9] O.A. von Lilienfeld, R. Ramakrishnan, M. Rupp, A. Knoll, Fourier series of atomic radial distribution functions: a molecular fingerprint for machine learning models of quantum chemical properties, Int. J. Quantum Chem. 115 (16) (2015) 1084–1093.
[10] K.T. Schütt, H. Glawe, F. Brockherde, A. Sanna, K.R. Müller, E.K.U. Gross, How to represent crystal structures for machine learning: towards fast prediction of electronic properties, Phys. Rev. B 89 (2014) 205118.
[11] A. Lopez-Bezanilla, O.A. von Lilienfeld, Modeling electronic quantum transport with machine learning, Phys. Rev. B 89 (2014) 235411.
[12] A.P. Bartók, M.J. Gillan, F.R. Manby, G. Csányi, Machine-learning approach for one- and two-body corrections to density functional theory: applications to molecular and condensed water, Phys. Rev. B 88 (2013) 054104.
[13] B. Meredig, A. Agrawal, S. Kirklin, J.E. Saal, J.W. Doak, A. Thompson, K. Zhang, A. Choudhary, C. Wolverton, Combinatorial screening for new materials in unconstrained composition space with machine learning, Phys. Rev. B 89 (2014) 094104.
[14] T. Mueller, E. Johlin, J.C. Grossman, Origins of hole traps in hydrogenated nanocrystalline and amorphous silicon revealed through machine learning, Phys. Rev. B 89 (2014) 115202.
[15] M. Fernandez, P.G. Boyd, T.D. Daff, M.Z. Aghaji, T.K. Woo, Rapid and accurate machine learning recognition of high performing metal organic frameworks for CO_2 capture, J. Phys. Chem. Lett. 5 (17) (2014) 3056–3060.
[16] L. Ward, A. Agrawal, A. Choudhary, C. Wolverton, A general-purpose machine learning framework for predicting properties of inorganic materials, npj Comput. Mater. 2 (2016) 16028.
[17] Y. Liu, T. Zhao, W. Ju, S. Shi, Materials discovery and design using machine learning, J. Materiomics 3 (3) (2017) 159–177.
[18] K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O.A. Von Lilienfeld, K.-R. Müller, A. Tkatchenko, Machine learning predictions of molecular properties: accurate many-body potentials and nonlocality in chemical space, J. Phys. Chem. Lett. 6 (12) (2015) 2326.
[19] B. Meredig, C. Wolverton, Dissolving the periodic table in cubic zirconia: data mining to discover chemical trends, Chem. Mater. 26 (6) (2014) 1985–1991.
[20] S.G. Javed, A. Khan, A. Majid, A.M. Mirza, J. Bashir, Lattice constant prediction of orthorhombic abo 3 perovskites using support vector machines, Comput. Mater. Sci 39 (3) (2007) 627–634.
[21] B. Revard, W. Tipton, R. Hennig, Structure and stability prediction of compounds with evolutionary algorithms, in: S. Atahan-Evrenk, A. Aspuru-Guzik (Eds.), Prediction and Calculation of Crystal Structures, Topics in Current Chemistry, Springer International Publishing, vol. 345, 2014, pp. 181–222.
[22] Genetic algorithm for structure and phase prediction, 2014. <http://gasp.mse.ufl.edu/>.
[23] W.W. Tipton, C.A. Matulis, R.G. Hennig, Ab initio prediction of the li5ge2 zintl compound, Comput. Mater. Sci. 93 (2014) 133–136.
[24] M. Ashton, J. Paul, S.B. Sinnott, R.G. Hennig, Topology-scaling identification of layered solids and stable exfoliated 2d materials, Phys. Rev. Lett. 118 (10) (2017) 106101.
[25] K. Mathew, A.K. Singh, J.J. Gabriel, K. Choudhary, S.B. Sinnott, A.V. Davydov, F. Tavazza, R.G. Hennig, Mpinterfaces: A materials project based python tool for high-throughput computational screening of interfacial systems, Comput. Mater. Sci. 122 (2016) 183–190.
[26] S.P. Ong, W.D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V.L. Chevrier, K.A. Persson, G. Ceder, Python materials genomics (pymatgen): a robust, open-source python library for materials analysis, Comput. Mater. Sci. 68 (2013) 314–319.
[27] J. Behler, Atom-centered symmetry functions for constructing high-dimensional neural network potentials, J. Chem. Phys. 134 (7) (2011) 074106.
[28] A.P. Bartók, R. Kondor, G. Csányi, On representing chemical environments, Phys. Rev. B 87 (18) (2013) 184115.
[29] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O.A. Von Lilienfeld, A. Tkatchenko, K.-R. Müller, Assessment and validation of machine learning methods for predicting molecular atomization energies, J. Chem. Theory Comput. 9 (8) (2013) 3404–3419.
[30] G. Montavon, K. Hansen, S. Fazli, M. Rupp, F. Biegler, A. Ziehe, A. Tkatchenko, A.V. Lilienfeld, K.-R. Müller, Learning invariant representations of molecules for atomization energy prediction, in: Advances in Neural Information Processing Systems, 2012, pp. 440–448.
[31] O. Isayev, D. Fourches, E.N. Muratov, C. Oses, K. Rasch, A. Tropsha, S. Curtarolo, Materials cartography: representing and mining materials space using structural and electronic fingerprints, Chem. Mater. 27 (3) (2015) 735–743.
[32] Y. Wang, J. Lv, L. Zhu, Y. Ma, Calypso: A method for crystal structure prediction, Comput. Phys. Commun. 183 (10) (2012) 2063–2070.
[33] F. Faber, A. Lindmaa, O.A. von Lilienfeld, R. Armiento, Crystal structure representations for machine learning models of formation energies, Int. J. Quantum Chem. 115 (16) (2015) 1094–1101.
[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[35] B. Schölkopf, A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, 2002.
[36] A.J. Smola, B. Schölkopf, A tutorial on support vector regression, Stat. Comput. 14 (3) (2004) 199–222.
[37] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (1) (2012) 281–305.