









RESEARCH ARTICLE | MAY 28 2024

Discovering melting temperature prediction models of inorganic solids by combining supervised and unsupervised learning

Vahe Gharakhanyan ; Luke J. Wirth ; Jose A. Garrido Torres ; Ethan Eisenberg ; Ting Wang; Dallas R. Trinkle ; Snigdhanu Chatterjee ; Alexander Urban  



J. Chem. Phys. 160, 204112 (2024)

<https://doi.org/10.1063/5.0207033>



09 August 2024 21:27:38



Nanotechnology &
Materials Science



Optics &
Photonics



Impedance
Analysis



Scanning Probe
Microscopy



Sensors



Failure Analysis &
Semiconductors



Unlock the Full Spectrum.
From DC to 8.5 GHz.

Your Application. Measured.

[Find out more](#)



Discovering melting temperature prediction models of inorganic solids by combining supervised and unsupervised learning

Cite as: J. Chem. Phys. 160, 204112 (2024); doi: 10.1063/5.0207033

Submitted: 5 March 2024 • Accepted: 29 April 2024 •

Published Online: 28 May 2024



Vahe Gharakhanyan,^{1,2} Luke J. Wirth,³ Jose A. Garrido Torres,⁴ Ethan Eisenberg,⁴ Ting Wang,⁴ Dallas R. Trinkle,³ Snigdhasu Chatterjee,⁵ and Alexander Urban^{2,4,a)}

AFFILIATIONS

¹ Department of Applied Physics and Applied Mathematics, Columbia University, New York, New York 10027, USA

² Columbia Electrochemical Energy Center, Columbia University, New York, New York 10027, USA

³ Department of Materials Science and Engineering, University of Illinois Urbana-Champaign, Urbana, Illinois 61801, USA

⁴ Department of Chemical Engineering, Columbia University, New York, New York 10027, USA

⁵ School of Statistics, University of Minnesota, Minneapolis, Minnesota 55455, USA

^{a)} Author to whom correspondence should be addressed: a.urban@columbia.edu

ABSTRACT

The melting temperature is important for materials design because of its relationship with thermal stability, synthesis, and processing conditions. Current empirical and computational melting point estimation techniques are limited in scope, computational feasibility, or interpretability. We report the development of a machine learning methodology for predicting melting temperatures of binary ionic solid materials. We evaluated different machine-learning models trained on a dataset of the melting points of 476 non-metallic crystalline binary compounds using materials embeddings constructed from elemental properties and density-functional theory calculations as model inputs. A direct supervised-learning approach yields a mean absolute error of around 180 K but suffers from low interpretability. We find that the fidelity of predictions can further be improved by introducing an additional unsupervised-learning step that first classifies the materials before the melting-point regression. Not only does this two-step model exhibit improved accuracy, but the approach also provides a level of interpretability with insights into feature importance and different types of melting that depend on the specific atomic bonding inside a material. Motivated by this finding, we used a symbolic learning approach to find interpretable physical models for the melting temperature, which recovered the best-performing features from both prior models and provided additional interpretability.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0207033>

I. INTRODUCTION

The compound melting temperature is an essential thermodynamic quantity for many processes, for example, for developing electrolytes for the electrolytic recycling of metals from spent electronics^{1–15} and for smelting processes to extract a base metal from its ore.^{16–19} The melting point can be measured directly or obtained from phase diagrams when available for the chemical compositions in question, which is often not the case for non-metallic inorganic materials with high melting temperatures. For a large portion of inorganic chemical space, currently, no well-curated non-commercial public melting-temperature databases exist, to the author's knowledge.

Motivated by the challenges in the experimental measurement of melting temperatures, various theoretical and computational methods for estimating melting temperatures have been proposed. We distinguish between four general techniques of attaining melting temperatures: (a) pure theoretical approaches based on theories that apply to specific classes of materials,^{20–25} (b) direct molecular dynamics (MD) simulations of melting either based on first-principles calculations^{26–35} or empirical interatomic potentials,^{36–38} (c) thermodynamic models based on the CALPHAD (CALculation of PHase Diagrams) method,^{39,40} and (d) machine learning (ML) models for the direct prediction of melting temperatures without simulation.^{41–46}

Theories describing melting have been mostly limited to metals and metallic alloys. The first such theory, the Lindemann melting criterion, proposed by Frederick Lindemann in 1910,²⁰ relates melting to the displacement of atoms from their equilibrium lattice sites: as temperature rises, the average amplitude of thermal vibrations in the lattice increases, and melting begins when the vibrations become severe enough that atoms collide with surrounding atoms. Lindemann used a closed-form equation to connect the melting point to either the Debye frequency or the Debye temperature. This criterion has been demonstrated to apply to single-component metal and metalloid systems,²⁰ but when the system's complexity increases, such as in the case of high-entropy alloys, modifications to this melting rule are required.^{47–51} The Lindemann criterion does not generally apply to non-metallic compounds.

Direct molecular dynamics (MD) melting simulations are challenging with first-principles methods, such as density-functional theory (DFT),^{52,53} because of the long timescales required at temperatures close to the melting point. However, various accelerated MD-based first-principles techniques have been proposed, such as the large-size coexistence method^{37,54–56} and the fast-heating method.⁵⁷ DFT-based schemes have also been used to calculate the Gibbs/Helmholtz free energy values of solid and liquid phases and then solve for the melting temperature by equating these thermodynamic quantities for separate phases.^{58,59} Despite such acceleration techniques, first-principles approaches are computationally demanding and limited to comparatively small simulation cells and timescales, reducing the accuracy of melting-temperature predictions. The computational cost of MD simulations is far lower with interatomic potentials, but accurate potentials that can describe both the crystal and the melt of an inorganic material are often unavailable.⁶⁰

Thermodynamic models based on the CALPHAD approach can yield highly accurate phase diagrams, including melting transitions.^{61,62} Unfortunately, few CALPHAD models of inorganic materials have been published, so model availability is a limiting factor.

The methodologies reviewed above are restricted to certain compound classes and are constrained by method accuracies and a lack of experimental data. These limitations motivated the development of techniques that combine ML models with physics-based models for melting-point prediction. Seko *et al.* demonstrated that incorporating DFT features into element-based materials fingerprints improves the predictive power of ML models for predicting melting temperatures of binary and single-component solid materials.⁴¹ The authors found that support vector regression led to the best melting-temperature prediction model. Guan and Viswanathan developed an artificial neural network with Bayesian optimization for hyperparameter tuning that could successfully learn the melting temperatures of alloy systems using melting data from CALPHAD.⁴² Hong and co-workers demonstrated melting temperature prediction for an extensive dataset of more than 9300 materials utilizing a graph neural network for materials embedding from compositions and a residual neural network for melting temperature prediction.^{63–65} The dataset used in their work mostly contained compounds with melting temperatures below 2000 K (about 90% of the dataset). As seen from the reviewed examples, the prior work on combining ML with first principles for melting temperature prediction was

generally limited to certain types of materials and/or narrow ranges of melting temperatures.

In the present study, we address the limitations of melting temperature models through a combination of supervised and unsupervised learning. Our focus lies on binary inorganic solids with melting temperatures up to 4000 °C because of the tremendous technological relevance^{9,12} of this compound space. We particularly focused on binary inorganic materials because of the lack of experimental data and models. However, we made sure that our featurization methods are transferable to more complex compositions such that, given the availability of an extended dataset of more complex materials, new ML models can be fitted without the need to revise the materials embedding method. We considered additional unsupervised learning tasks to study the physics behind melting, motivated by the observation that melting is directly related to the bonding types within a material, and different bond-related features may affect melting differently (Lindemann criterion⁴⁷). To this end, we employed unsupervised learning to partition our dataset into separate groups based on bonding within materials. To learn more about the underlying physics of melting and approach interpretability, we used symbolic regression^{66,67} to learn a closed-form equation for melting.

II. METHODS

A. Model development

To develop a machine learning model to predict the melting temperatures of non-metallic, inorganic solids, we followed an approach consisting of the following five steps: (1) *data collection*, where we compiled a dataset of experimentally measured melting temperatures from various sources; (2) *featurization*, where we explored a range of features and used domain knowledge to select the most appropriate ones for representing the materials; (3) *model selection and training/construction*, where we evaluated several machine learning models on the featurized data to identify the best-performing ones; (4) *model evaluation*, where we tested the performance of the best model on an independent dataset to ensure generalizability; and (5) *result interpretation*, where we constructed symbolic regression models to understand the underlying relationships better. Detailed descriptions of each step are provided in the following Secs. II A 1–II A 5.

1. Melting-temperature data

We compiled a dataset of 476 binary ionic compound melting temperatures spanning a temperature range of 0–4000 °C. Figure S1 provides violin plots that show the distributions of these melting points over the various partitions of our dataset. The experimental melting temperature data were primarily obtained from two CRC handbooks.^{68,69} Compounds in our dataset are exclusively of the metal–nonmetal, metal–metalloid, and metalloid–nonmetal element combinations, with metal–metal (alloys with predominantly metallic bonding), metalloid–metalloid, and nonmetal–nonmetal (with mostly covalent bonding) compound types excluded. We imposed the constraint on compound types since we are primarily interested in ionic compounds and minerals. Note that the records of compounds with high melting points (>2000 °C) often report ranges of temperatures rather than single points. In addition, melting temperature data from different sources occasionally had conflicting

values. In these cases, we include the average reported melting temperature as one single point in our dataset for a given compound.

We partitioned our melting temperature dataset into training and test sets for ML model training. The first set was used to build models, while the second was only used to evaluate them. We included those materials in the test set whose elastic moduli were not available in the Materials Project database^{70,71} and were calculated by ourselves (see Sec. II C for details). This makes the test set performance especially meaningful since the model will, in practice, be primarily evaluated for compounds that have not previously been characterized, and thus, the elastic moduli values will have to be manually calculated. This resulted in a 420/56 (88%/12%) training/test split.

Figure S2 illustrates the elemental distributions within the training and test sets. In terms of the statistical distribution of element types, the test set is representative of the training set. It is apparent from the melting temperature distributions that the test set is biased to lower melting temperatures. We consider this acceptable since the experimental values for high melting temperatures are generally subject to significant uncertainties.

TABLE I. Compound and elemental features, their symbols, and units.

Property of	Feature	Symbol	Unit
Compound	Cohesive energy	E_{coh}	eV/atom
	Formation energy	H	eV/atom
	Bulk modulus	K	GPa
	Shear modulus	G	GPa
	Density	ρ	g/cm ³
	Bond ionic character	%IC	...
	Molar mass	M	g/mol
	Atomic number	Z	...
Element	Atomic radius	R	Å
	Molar volume	V_{m}	m ³ /mol
	Pauling electronegativity	EN	...
	Periodic row number	N_{R}	...
	Periodic group number	N_{G}	...
	Elemental melting temperature	T_{m}	K

2. Feature construction and selection

We based our materials representation on a combination of *compound features*, i.e., properties of a given compound, and *element-based features*, i.e., features built from elemental properties using different statistical moments and averaging approaches, as shown in Table I.

Except for the bond ionic character, the compound features were obtained from DFT calculations extracted from the Materials Project⁷² using the most stable entry for the given composition, i.e., the entry with energy on or closest to the lower convex hull of formation energies. We performed DFT calculations of the bulk and shear moduli (see Sec. II C for details) for those compounds for which elastic moduli values were unavailable in the Materials Project (or were only available as ML predictions)⁷³ and included these compounds in the independent test set. The bond ionic character %IC was originally suggested by Pauling⁷⁴ as a measure of the percentage of ionicity in the bonds between two atoms A and B and is constructed from elemental electronegativities as %IC = $\left(1 - \exp\left[\frac{(EN_{\text{A}} - EN_{\text{B}})^2}{4}\right]\right) \cdot 100\%$, where EN_i is the electronegativity of element i .

To obtain compound material features from elemental properties, we employed five statistical moments and averaging approaches, either accounting for stoichiometry/composition or only based on the combination of elements: the arithmetic average, standard deviation, harmonic average, quadratic average, and geometric average (see Table II). These moment and averaging methods were chosen to ensure that the features are symmetric with respect to the permutation of atomic species. The features additionally generalize without modifications to simpler unary systems and more complex systems (ternary, quaternary solids, or higher-order compositions).

Overall, we obtained 86 features (six compound features: five DFT features from the Materials Project and the bond ionic character, and 80 composed features from eight elemental features with five statistical moment and averaging methods and both composition-based and composition-agnostic weighting). This means our materials fingerprint encodes elemental properties, compositions, and compound-specific physical properties. We note that the inclusion of DFT-based features makes it possible to distinguish between

TABLE II. Statistical moments and averaging methods for elemental feature Z and compound A_xB_y .

Method	Notations for composition-based and composition-agnostic features, respectively	Equations for composition-based features (for composition-agnostic features, $x = y = 1$)
Arithmetic average	$\langle Z \rangle_{\text{aw}}$ and $\langle Z \rangle_{\text{a}}$	$\frac{xZ_{\text{A}} + yZ_{\text{B}}}{x + y}$
Standard deviation	$\langle Z \rangle_{\text{sw}}$ and $\langle Z \rangle_{\text{s}}$	$\frac{ xZ_{\text{A}} - yZ_{\text{B}} }{x + y}$
Harmonic average	$\langle Z \rangle_{\text{hw}}$ and $\langle Z \rangle_{\text{h}}$	$\frac{x + y}{\frac{x}{Z_{\text{A}}} + \frac{y}{Z_{\text{B}}}}$
Quadratic average	$\langle Z \rangle_{\text{qw}}$ and $\langle Z \rangle_{\text{q}}$	$\sqrt{\frac{xZ_{\text{A}}^2 + yZ_{\text{B}}^2}{x + y}}$
Geometric average	$\langle Z \rangle_{\text{gw}}$ and $\langle Z \rangle_{\text{g}}$	$\sqrt[x+y]{Z_{\text{A}}^x Z_{\text{B}}^y}$

polymorphs, such as, for instance, rutile and anatase titanium dioxide.

Applying these different averaging methods to our set of eight elemental features led to the creation of expectedly correlated features, as shown by a Pearson correlation coefficient analysis in Fig. S3. We provided our model with all of these features to investigate which averaging method would perform best. Furthermore, the best-performing models were mostly ensemble tree-based, and based on our experience, even minor non-collinearity in features can result in better ensemble tree-based models.

3. Supervised-learning model selection and evaluation

We considered different regression models: (a) a linear regression model for benchmark purposes, as implemented in the *scikit-learn* Python package;⁷⁵ (b) Gaussian process regression, kernel ridge regression, and support vector regression models (*scikit-learn*); (c) a vanilla light gradient boosting machine (LGBM) and a standard LGBM regressor, as implemented in *lightGBM*;⁷⁶ and (d) a random forest (RF) model, LGBM RF, as implemented in the *xgboost*⁷⁷ and *lightGBM* Python packages, respectively.

The performance of the models was measured by several error metrics: the mean absolute error (MAE), the root-mean-square error (RMSE), the mean absolute percentage error (MAPE), and the coefficient of determination (R^2). These metrics were computed for the training set with fivefold (fivefold CV), leave-one-out cross-validation (LOO-CV), and the test set. We constructed a hyperparameter grid for all the models, and hyperparameters were tuned with grid⁷⁸ and random search⁷⁹ techniques with fivefold cross-validation, as implemented in *scikit-learn*. Although there are too many parameters to carry out an exhaustive grid search easily, each parameter was manually adjusted, and the parameters that reduced the (average) RMSE over the fivefold cross-validation process were retained.

We tested several possibilities as the model output, including direct melting temperatures (in both K and °C units), logarithmic (to base 10) melting temperature (in K), and melting temperature from Vegard's law as $T_{m,Vegard} = T_m - \sum_i^{\text{Elements}} x_i T_m^{(i)}$, where $T_{m,Vegard}$, T_m , $T_m^{(i)}$, and x_i are the melting temperature from Vegard's law, the melting temperature of the compound, as well as the melting temperature and atomic fraction of element i , respectively. We also considered different data normalization techniques on the input features, specifically, standardization and min-max scaling as implemented in *scikit-learn*. Models with the best performance often resulted from training using standardized features as inputs and the logarithm of the melting temperatures as outputs.

4. Unsupervised-learning model construction

For the clustering of the materials into different compound groups, we performed k-means clustering⁸⁰ using a subset of hand-picked features related to the bond ionicity (bond ionic character and weighted arithmetic average and weighted standard deviation of electronegativity, periodic row, and periodic group numbers). The optimal number of clusters was determined by simultaneous Silhouette score⁸¹ and Calinski–Harabasz score⁸² analyses when considering cases of up to ten clusters. Silhouette analysis shows the separation distance between the resulting clusters (how similar a data point is to its own cluster compared to other clusters using

Euclidean distance). The Calinski–Harabasz score is the ratio of the sum of between-clusters dispersion and inter-cluster dispersion for all clusters and measures the separation between clusters based on Euclidean distances and the compactness of data points within each cluster. The Silhouette score and Calinski–Harabasz score analyses are presented in Figs. S4–S6, along with the clustering results projected on a two-dimensional space with principal component analysis (PCA) in Fig. S7.⁸³

With the combined unsupervised/supervised model, predicting the melting temperature of a material from the test set involves two steps. First, the cluster to which the test point belongs is identified using the (unsupervised) k-means model, and then, the regression model for the selected cluster is used to make the temperature prediction.

5. Symbolic regression model construction and validation

We employed the variable-selection assisted sure independence screening and sparsifying operator (VS-SISSO)⁶⁷ to construct physically interpretable descriptors of the melting temperatures. VS-SISSO composes derived features from a set of input physical features using a set of operators. The present work uses the $\{x_1 + x_2, x_1 - x_2, |x_1 - x_2|, x_1 \times x_2, x_1 \div x_2, \exp(x), \exp(-x), \ln(x), x^{-1}, x^2, x^3, \sqrt{x}, \sqrt[3]{x}\}$ set of operators, where x_i can be either an input or a previously derived feature. Models take the form $T_m = c_0 + c_1 X_1 + \dots + c_D X_D$, where c_i are linearly optimized coefficients that minimize the training RMSE, given a combination of D derived features X . Each model has a dimensionality D , i.e., the number of derived features that the model depends on, and a feature complexity f , which is the maximal number of operations used within one derived feature.

The iterative variable selection of VS-SISSO (compared to regular SISSO)⁶⁶ allows for efficient screening of large feature spaces by considering subsets (S) of the entire feature space that combine new features (S_a) with those of the best-performing models (S_b). Here, model construction uses nine different sets of hyperparameters where D and f take on values of 1, 2, or 3. S_a introduces four physical features at each step, S has a maximal size of 16, and model construction is considered converged when no improvement in training RMSE is seen over 50 iterations. Up to 10 000 of the best models are output from each calculation, and for 1D, 2D, and 3D models, the SIS-selected subspace of derived features contains a maximum of 100 000, 10 000, and 1000 items, respectively.

B. SHAP analysis

SHAP (SHapley Additive exPlanations) is a model interpretability method that uses Shapley values from game theory to estimate the predictive importance (SHAP score) of the features of a model.⁸⁴ Note that the SHAP method explains the ML model and not the data: SHAP importance is not a measure of the importance of a given feature according to the physics of a problem, but it instead shows how important a feature is to a given ML model.

We calculated the SHAP values to understand the predictions of the constructed models. The *shap* Python module⁸⁵ was used to estimate these values. SHAP values are generated for each model feature and have the same unit as the anticipated target (melting temperature or its derivatives). In our case, the best models resulted

from training on logarithmic melting temperatures and, hence, the scale of the SHAP values. Furthermore, SHAP values are additive, meaning that adding all the SHAP values for a specific prediction plus a base value (the mean of the target melting temperature values of the dataset) yields the model prediction value. For example, a SHAP value of 0.5 indicates that adding that particular feature value to which the SHAP value corresponds will increase the logarithmic melting temperature by 0.5 units from the base (mean) value.

We visualize SHAP analysis in the form of layered violin plots. The ten most important features are displayed in decreasing order of importance from top to bottom in these charts. The higher the importance, the greater the influence of that feature on the melting temperature prediction for the given model. For visualization purposes, for each SHAP value, the lighter colors (lower feature values) are closer to the horizontal line, while the darker colors (higher feature values) are closer to the edges of the layered violin plot.

C. Elastic moduli computations with DFT

A well-converged stress tensor is essential to derive correct elastic constants, such as bulk and shear moduli, from DFT. All DFT results reported in the present work were based on the projector augmented wave (PAW) method^{36,87} as implemented in the Vienna *Ab Initio* Simulation Package (VASP).^{26,88,89} We used the exchange–correlation functional by Perdew, Becke, and Ernzerhof (PBE).⁹⁰ The plane-wave energy cutoff was 700 eV, and the k-point density was 7000 pra (per reciprocal atom). Energies converged to less than 0.01 meV. The atomic positions and lattice constants were relaxed until the residual forces became less than 0.001 eV/Å. When the elastic tensor or the ionic-relaxation step failed to converge, the computation was restarted, but this time with new DFT parameters. As a result, the numerical parameters reported above were typical of many of our computations, but in certain circumstances, other

parameters were employed (energy cutoff of 520 eV and/or k-point density of 1000 pra).

To ensure compatibility with the data gathered from the Materials Project,^{70,71} calculations of the moduli in this work take the Voigt–Reuss–Hill (VRH) average form⁹¹ of the bulk modulus K_{VRH} and the shear modulus G_{VRH} calculated as

$$K_{VRH} = \frac{(K_V + K_R)}{2} \quad \text{and} \quad G_{VRH} = \frac{(G_V + G_R)}{2},$$

where the subscripts V and R indicate the upper (Voigt) and lower (Reuss) bounds, respectively, defined by

$$9K_V = (c_{11} + c_{22} + c_{33}) + 2(c_{12} + c_{23} + c_{31}),$$

$$\frac{1}{K_R} = (s_{11} + s_{22} + s_{33}) + 2(s_{12} + s_{23} + s_{31}),$$

$$15G_V = (c_{11} + c_{22} + c_{33}) - (c_{12} + c_{23} + c_{31}) + 3(c_{44} + c_{55} + c_{66}),$$

$$\frac{15}{G_R} = 4(s_{11} + s_{22} + s_{33}) - 4(s_{12} + s_{23} + s_{31}) + 3(s_{44} + s_{55} + s_{66}).$$

In the above equations, c_{ij} and s_{ij} are the elements of the stiffness and compliance tensors, respectively.

III. RESULTS

A. One-step approach: Direct regression models

We first used direct supervised learning methods to construct models for melting temperature prediction by training on the entire training set. The best model resulted from an ensemble gradient-boosted tree regression (GBR) with a feature set that included

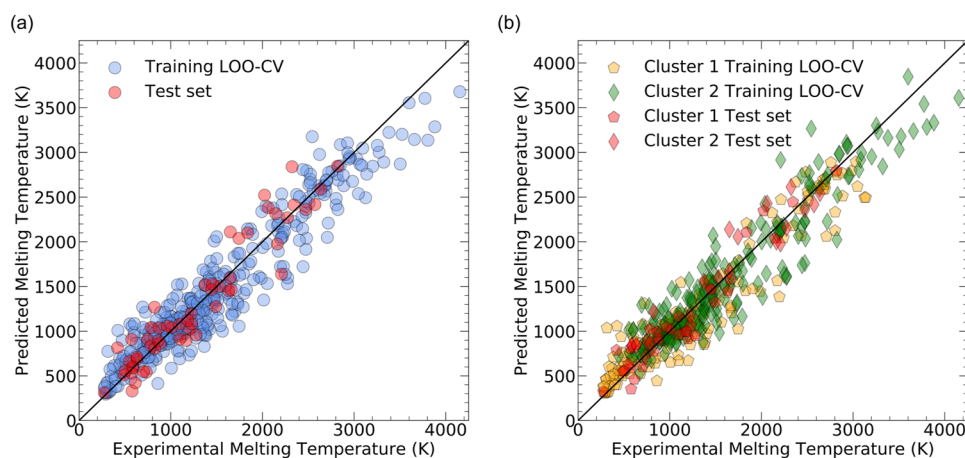


FIG. 1. Correlation plots between the experimental and predicted melting temperatures for the best models trained on the (a) whole training dataset and (b) training sets of clusters 1 and 2 separately. The training LOO-CV and test set labels represent ensemble averaged predictions over ten seeds. Table III shows the reported error metrics. The feature set used for training the model in (a) and for cluster 1 in (b) included compound features and both composition-based and composition-agnostic harmonic and quadratic average features of elemental properties. The feature set used for training the model for cluster 2 in (b) included compound features and both composition-based and composition-agnostic harmonic and arithmetic average features of elemental properties. The best model for the one-step approach in (a) resulted from the gradient-boosted tree regression, while random forest regression was the best model for clusters 1 and 2 in (b).

TABLE III. Error metrics for the best models for both the one-step approach and individual clusters in the two-step approach. We report LOO-CV and fivefold CV scores for the training set. We report MAE, RMSE, MAPE, and R^2 metrics. All scores listed are the average metrics for models across ten different seeds. For MAE and RMSE, besides the average metric, standard deviations are also listed. For the two-step approach, we also report a metric evaluated on the combination of clusters 1 and 2 to facilitate the comparison with the one-step approach.

Approach	Cluster	MAE (K)	RMSE (K)	MAPE (%)	R^2
Training set (LOO-CV)					
One-step	...	177 ± 1	243 ± 2	14	0.91
Two-step	1	169 ± 2	246 ± 4	17	0.89
Two-step	2	198 ± 1	267 ± 1	13	0.89
Two-step	Combined	184 ± 2	257 ± 3	15	0.89
Training set (fivefold CV)					
One-step	...	201 ± 1	266 ± 2	17	0.89
Two-step	1	189 ± 3	264 ± 4	19	0.87
Two-step	2	211 ± 2	276 ± 2	14	0.88
Two-step	Combined	201 ± 2	271 ± 3	17	0.88
Test set					
One-step	...	165 ± 4	218 ± 7	15	0.89
Two-step	1	136 ± 5	169 ± 6	15	0.92
Two-step	2	189 ± 7	229 ± 6	16	0.90
Two-step	Combined	156 ± 4	194 ± 4	15	0.92

compound properties and harmonic and quadratic (composition-based and composition-agnostic) averages of elemental properties [Fig. 1(a)]. We performed ensemble averages using ten different random seeds (statistics are presented in Fig. S8), and the error metrics are presented in Table III. Figure 1(a) shows the correlation of the predicted melting temperatures from the best

model of the one-step approach with the reference experimental values. Figure 1(a) shows that the predictions strongly correlate with the actual values and are scattered around the perfect diagonal without a systematic error. Figure S9 shows that the distribution of the errors for the test set is similar to that of the training set, further indicating that the model is not overfitted. An

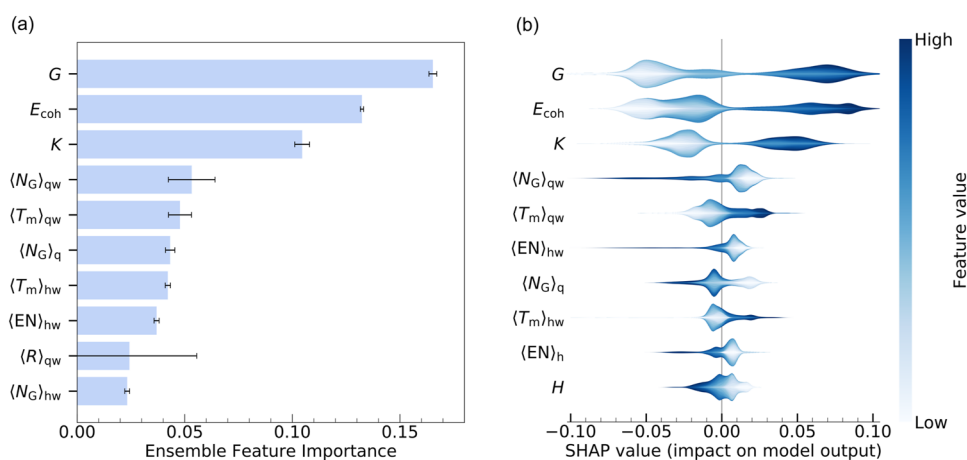


FIG. 2. Feature importance for the one-step approach and the best ensemble model trained on the whole training dataset computed using (a) the ensemble tree method and (b) the SHAP analysis. The features are ranked in order of their importance from top to bottom. Only the top ten most important features are present for both approaches. The ensemble tree importance analysis in (a) represents ensemble-averaged results over ten seeds, and the horizontal error bars indicate values of importance one standard deviation above and below their respective average values over ten different seeds. In (b), a positive/negative SHAP value shows that the feature increases/decreases the melting temperature prediction. For each SHAP value in (b), the lighter colors (lower feature values) are closer to the horizontal line, while the darker colors (higher feature values) are closer to the edges of the layered violin plot.

overview of the performance of other considered models is shown in Table S1.

Figure 2 represents feature importance through ensemble feature importance⁹² and SHAP analysis.⁸⁴ Both feature importance methods identified the cohesive energy and elastic moduli features as the most important.

B. Two-step approach: Combined clustering and regression models

The feature-importance analysis of Fig. 2 identified the shear modulus and cohesive energy as the two most important features for melting temperature prediction with a tree-based model. As reviewed in the Introduction, melting occurs when the binding between atoms is partially overcome, so the importance of cohesive energy also agrees with intuition. The binding between atoms in a compound is due to different interactions, most importantly, covalent, ionic, and metallic bonding. The nature of the bonding determines, for example, how the cohesive energy changes with the coordination number. For instance, in ionic compounds, the cohesive energy is proportional to the coordination number, while it behaves as the square root of the coordination number in metals.

Motivated by these considerations, we investigated a second class of models involving an additional clustering step before the regression, in which materials were grouped based on features related to the bonding character. For this two-step approach, we first used an unsupervised-learning method to separate our dataset into clusters and then fit supervised learning models within each cluster separately. For melting temperature prediction, first, the cluster a material belongs to is identified, and then, the corresponding regression model is used for the melting point prediction.

Our clustering approach was based on k-means clustering.⁸⁰ Clustering was based on features that we intuitively associated with the bond ionicity (bond ionic character and weighted arithmetic average and weighted standard deviation of electronegativity,

periodic row, and periodic group numbers) to facilitate the interpretation of different melting behaviors based on bonding in our dataset. The optimal number of clusters was chosen from simultaneous Calinski–Harabasz score⁸² (Fig. S4) and silhouette score⁸¹ (Figs. S5 and S6) analyses, both of which resulted in an optimal number of two clusters. Clusters 1 and 2 contained 198 and 222 compounds, respectively. The clustering results projected on a two-dimensional space with principal component analysis (PCA) are presented in Fig. S7.

Within each cluster, we fitted regression models for melting temperature prediction. The best models resulted from an ensemble random forest regression (RFR) for both clusters with feature sets that included compound properties and (both composition-based and composition-agnostic) harmonic and quadratic averages of elemental properties for cluster 1 and compound properties and (both composition-based and composition-agnostic) harmonic and arithmetic averages of elemental properties for cluster 2 [Fig. 1(b)]. We performed ensemble averages over ten different seeds (statistics are presented in Fig. S8), and the error metrics are presented in Table III. An overview of the performance of other considered models is shown in Table S1.

We carried out a feature importance analysis through the ensemble feature importance (Fig. S10) and SHAP method for both clusters (Fig. 3).

To compare the results from the two-step approach to those from direct supervised learning of Sec. III A, we evaluated the overall error scores for the complete dataset, including both clusters (Table III and Fig. S11).

C. Symbolic models from VS-SISSO

While our model evaluation found tree ensemble methods to perform best for the melting temperature dataset, these models are not readily interpretable. This makes it challenging to determine

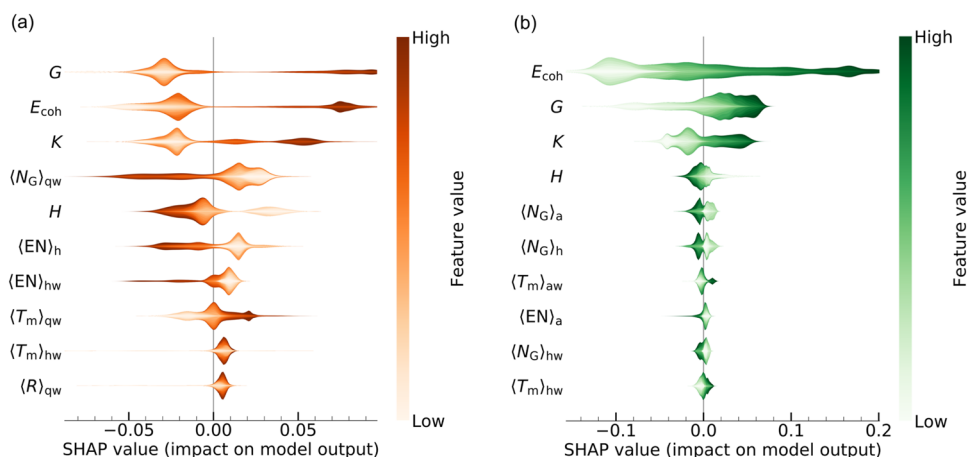


FIG. 3. Feature importance analysis from the SHAP method for (a) cluster 1 and (b) cluster 2. The features are ranked in order of their importance from top to bottom. Only the top ten most important features are present for both clusters. A positive/negative SHAP value shows that the feature increases/decreases the melting temperature prediction. For each SHAP value, the lighter colors (lower feature values) are closer to the horizontal line, while the darker colors (higher feature values) are closer to the edges of the layered violin plot.

TABLE IV. VS-SISSO-generated models for melting temperature for the one- and two-step approaches with the lowest average RMSE scores of the training and test sets of all generated 1D models with feature complexity $f = 3$ using all available features. Two models are provided for the entire training set and cluster 2 cases since providing SISSO with the different types of averaging methods yields slightly better performance. For cluster 1, the best models contain only the arithmetic averages. Here, we report RMSE and R^2 scores for the training and test sets. For models constructed using dimensionless features and melting temperatures, consult Table S2.

Approach	Cluster	T_m (K) equation/model from VS-SISSO	Training set		Test set	
			RMSE (K)	R^2	RMSE (K)	R^2
One-step	...	$389 + 56(\sqrt[3]{G} \cdot E_{\text{coh}} \cdot \langle R \rangle_a)$	368	0.78	337	0.75
	...	$282 + 61(\ln(G) \cdot E_{\text{coh}} \cdot \langle R \rangle_{\text{qw}})$	342	0.81	310	0.79
Two-step	1	$276 + 58(\sqrt[3]{G} \cdot E_{\text{coh}} \cdot \langle R \rangle_a)$	358	0.76	324	0.69
	2	$730 + 129\left(\frac{E_{\text{coh}}^2}{\sqrt{\langle N_G \rangle_a}}\right)$	318	0.84	287	0.84
	2	$685 + 58\left(\frac{E_{\text{coh}}^2}{\sqrt{\langle EN \rangle_{\text{hw}}}}\right)$	310	0.85	290	0.83

in which way the materials in the two clusters of Sec. III B behave differently when melting.

Table IV provides VS-SISSO-generated symbolic expressions for the melting temperature based on the entire dataset as well as for the two clusters individually, giving qualitative insights into the melting physics that dominate within each one. Table S2 shows models with dimensionless features for dimensionless melting temperature model construction. Models selected for inclusion in Table IV and Table S2 have the lowest averaged training and test RMSEs of all generated 1D models with feature complexity $f = 3$ from the set of features that yields the best training RMSE.

IV. DISCUSSION

We constructed models for predicting the melting temperature of inorganic, non-metallic compounds using a direct one-step supervised learning approach and a two-step approach based on an additional unsupervised clustering step.

Fitting models without elastic moduli features yielded errors greater than 400 K, so we decided that incorporating and computing elastic features is necessary. We began our studies with a dataset of about 600 compounds. For high melting temperature compounds that lacked elastic moduli data in the Materials Project database, we computed elastic moduli ourselves but experienced numerical and convergence errors in some cases. This led us to exclude these compounds from our dataset, and as a consequence, very high melting temperatures are absent from the test set.

For the best-performing direct one-step regression model, the first and third most important features are the shear and bulk moduli, respectively. The elastic moduli relate to the resistance to elastic deformation when stress is applied to the compound. As such, these quantities are related to the bonding within the materials that determine how easily bonds within a material can be bent and eventually broken, which is an intuitive measure for melting temperatures. Intuition tells us that soft, deformable materials have a lower melting temperature than hard, brittle materials. The second most important feature, cohesive energy, is the difference between the bulk energy

and the sum of the energy of isolated atoms. It is also an intuitive feature since this energy needs to be partially overcome during melting.

While other top features, such as the statistical combinations of elemental periodic group number and elemental melting temperatures, are also important based on chemical intuition, our model cannot provide specific interpretable rules that relate these features to melting. This is a general limitation of black-box tree-based ML models since the decision criteria a model uses to generate predictions are often too complex to be interpreted directly. For this reason, we confirm our importance studies with SHAP analysis, which clearly shows the same ranking of feature importances for the five most important features for the model. For the first three most important features, elastic moduli and cohesive energy, SHAP analysis shows that high feature values tend to increase the predicted melting temperatures, while low feature values decrease the predicted melting temperature. This can also be seen in the correlation plot of the cohesive energy with the melting temperature (Fig. S12) and the correlation plot of the cohesive energy with the elastic moduli (Fig. S13). Higher cohesive energies result in higher melting temperatures.

Results from the two-step approach provide further insights. The distribution of elements within the two clusters (Fig. S14) shows that cluster 1 contains most halides and oxides, while cluster 2 contains B-, C-, N-, and O-group compounds except oxides. Given the types of materials that are included in our dataset and based on the elemental distributions, we conclude that cluster 1 incorporates more ionic compounds, while cluster 2 comprises less ionic and more covalent solid materials. The melting temperature distribution of the two clusters (Fig. S1b) shows that very-high (>3150 K) melting temperatures appear only in cluster 2. The compounds with extreme melting temperatures are mostly metal carbides known for their high melting temperatures resulting from a combination of inter- and intra-molecular covalent bonds that form covalent solid networks for this category of materials.^{93,94}

For cluster 1, the ranking and relative importances of the first four features are similar to those found in the one-step approach for the entire dataset. For cluster 2, though, we observe significant differences. By far, the most important feature for cluster 2 is cohesive

energy related to the melting temperature, as described above. In the SHAP analysis of Fig. 3, it is seen that very high cohesive energy values increase the predicted melting temperatures with a greater extent and, vice versa, very low cohesive energy values significantly decrease the predicted melting temperatures. While elastic moduli are the next important features for cluster 2, they have less impact on the model output when compared to the cohesive energy feature. Here, we conclude that cohesive energy across the entire range of values is crucial to melting point prediction for more covalent solid materials. We also see that, in general, other features are evenly split in terms of the feature value and SHAP value, meaning they enter the melting point prediction in more complex relationships.

Table III compares the predictive power of the one- and two-step approaches with tree ensemble methods. In terms of the training set LOO-CV scores, the MAE for cluster 2 is higher (by ~20 K) than that of the entire training set from the one-step approach, while the MAPE score is reduced because the fraction of very-high (>3150 K) melting temperature materials in cluster 2 is higher than that in cluster 1. The training set LOO-CV scores are very comparable among the one- and two-step approaches when computing the combined error metrics for clusters 1 and 2. The same can be noted for the training set fivefold CV scores. Note that we generally observe lower RMSE and MAE error metrics from the test set because it is limited to a lower melting temperature range, so this does not indicate underfitting. The main differences are noticeable in the test set metrics. Specifically, the test set MAE and RMSE for cluster 1 are lower than that of the one-step approach by 29 and 49 K, respectively. This also comes with the cost of increasing the aforementioned metric values for cluster 2. However, overall, when comparing the whole test set to the combination of clusters 1 and 2, we see only a slight improvement in predictive power compared to the one-step approach (reduction in MAE of 9 K and RMSE of 24 K). This is not surprising since decision-tree-based models can internally perform a feature-based clustering so that a one-step tree-ensemble model can effectively replace the combined k-means clustering and regression steps.

However, this is not the case for other models, such as linear regression models, and here, the difference between the one-step and two-step approaches is much more significant. From Table S1, linear models (also with regularization) always result in MAE scores of greater than 240 K and RMSE scores greater than 340 K. These scores are at least 100 K higher than the respective best-performing models reported in Table III and are also significantly higher (by about 30 K) than what is attainable by the two-step approach when using the same linear models. Note that the linear models can be considered a benchmark of what is attainable with the simplest models, but they achieve poor metrics compared to the non-linear models we have evaluated in this work (Table S1). We also attempted model stacking but decided not to report more complex and less interpretable models with only slightly better performance.

Since clustering is beneficial for linear regression, we expected it also to facilitate symbolic regression with SISSO. For the models trained on the whole dataset, the 11 best-performing 1D models with $f = 3$ with average train-test RMSEs of 326–382 K all involve combinations of the shear modulus (G), cohesive energy (E_{coh}), and the statistical combination of elemental atomic radii ($\langle R \rangle$). Model complexity can significantly affect feature selection; the top 15 1D

and $f = 2$ models for both clusters all include E_{coh} and the statistical combination of elemental electronegativities ($\langle \text{EN} \rangle$) while spanning a slightly higher range of average train-test RMSEs from 341 to 407 K. Larger models offer relatively limited gains in accuracy while becoming less parsimonious and therefore harder to interpret. For example, the top 3D model with $f = 3$ trained on the data from clusters 1 and 2 has training and test RMSE values of 312 and 299 K, respectively, comparable to the 1D values of 368 and 337 K. Another example, the best-performing 3D model with $f = 3$ trained on the cluster 2 takes the form $T_{\text{m}} = c_0 + c_1 \frac{E_{\text{coh}}^2}{\sqrt{\langle N_{\text{G}} \rangle}} + c_2 \frac{\langle \rho \rangle \cdot \langle \text{EN} \rangle \cdot \langle N_{\text{G}} \rangle}{\langle Z \rangle} + c_3 \frac{\exp(\langle \text{EN} \rangle)}{\sqrt[3]{\langle N_{\text{G}} \rangle}}$ and is significantly more complicated than the model shown in the bottom row of Table IV, but the averaged training and test RMSEs of this 3D model improve on those of the 1D one by only 22 K.

Comparing which physical features are commonly selected by SISSO across various model complexities can provide insights into which features are most essential to describing each cluster. The cohesive energy, E_{coh} , is present in all 1D models of Table IV, but it is often absent in cluster 1 models that perform nearly as well (e.g., with an average train and test RMSE of 377 K). On the other hand, the shear modulus, G , regularly appears in cluster 1 or one-step approach models but is never selected in a 1D cluster 2 model with $f = 2$. These patterns suggest that G is particularly important for describing the melting temperature of cluster 1 compounds, and E_{coh} is likewise the key for cluster 2. Additionally, as Table IV illustrates, cluster 2 models tend to have higher c_0 values. While this can lead to overestimating relatively low melting temperatures, cluster 2-only RMSEs are lower than in either case where cluster 1 compounds are present, which have intercepts near room temperature. We also investigated how exactly each feature enters the melting equation. In general, we noted that physical feature selection is more critical than operator choice for SISSO. Even though we noted that for ionic compounds, the cohesive energy is proportional to the coordination number, while it behaves as the square root of the coordination number in metals, as seen in Fig. S15, there appears to be no distinct mathematical operator that leads to decreased test set RMSE metric when acted on E_{coh} (although a few such as E_{coh}^{-1} rarely appear in well-performing simple models).

Indeed, with a combined average test-set RMSE of 304 K, weighted by the number of materials in each cluster, the two-step SISSO model is slightly better than the model fitted to the entire dataset with an RMSE of 310 K. However, the most interesting observation is that the data in cluster 2 can be represented well with symbolic models depending only on two features: the squared cohesive energy and the inverse square root of either the statistical combination of elemental periodic groups or the electronegativities (Table IV). The interchangeability of the periodic group and electronegativity is intuitive as the electronegativity strongly correlates with the group in the Periodic Table for cluster 2. The expression $T_{\text{m}} \approx a + b \frac{E_{\text{coh}}^2}{\sqrt{\langle \text{EN} \rangle_{\text{hw}}}}$ with $a = 685$ K and $b = 58$ K/eV² and the cohesive energy in eV can thus be used to obtain an estimate of the melting temperature of non-oxides and non-halides with melting points above 685 K. The symbolic regression studies also corroborate that the most important features are those picked up by feature importances and SHAP analysis of the tree ensemble models.

The linear model fitted on the entire 86-feature input results in the following metrics: for the training set LOO-CV—RMSE: 346 K, MAE: 259 K, for the training set five-fold CV—RMSE: 378 K, MAE: 289 K, and for the test set—RMSE: 435 K, MAE: 353 K. Similar error metrics are observed for both clusters 1 and 2. Because of the computing requirements, we did not carry out LOO-CV or fivefold CV studies for the VS-SISSO models. The test set RMSE scores for the linear models are higher than those from the VS-SISSO models with a difference of less than 100 K. This is a very notable advantage of using a symbolic learning model instead of a simple linear model, as even the least complex VS-SISSO models with small values of dimensionality (1D) and feature complexity (up to three considered in this work), such as those shown in Table IV, can result in better fittings to the data and provide interpretability, albeit at an increased burden of computational complexity.

To verify that the reported VS-SISSO models have errors that reflect the capabilities of symbolic machine learning, although less physically interpretable, we also generated 1D, $f = 3$ complexity models by providing VS-SISSO with initially dimensionless melting temperatures and features obtained either through feature normalization, standardization, or scaling, as proposed in work introducing the idea of dimensionless features to extend the search space of symbolic models.^{95,96} The dimensionless approach identifies new models for the dimensionless melting temperature containing the dimensionless bulk modulus, such as $T_m^a \approx 0.22 + 0.81(\sqrt[3]{K} \cdot E_{\text{coh}} \cdot \langle R \rangle_a)$ and $T_m^a \approx 1.57 + 0.57(K - 2\langle EN \rangle_a)$ for the training sets of the entire dataset and cluster 1, respectively. These models perform particularly well on the test set at the expense of the training set. The error metrics and feature selections of the best models from these dimensionless feature transformation approaches are shown in Table S2 and are comparable to those shown in Table IV, suggesting that the reported models are indeed representative.

Although the present work is aimed at simplicity and interpretability, it is educative to compare our results to previously published models. The best model reported by Seko *et al.* on their smaller dataset of 248 unary and binary materials was a support vector regression model using a feature set composed of compound properties from DFT and elemental properties.⁴¹ This model achieved a tenfold CV RMSE of 265 K for the training set and an RMSE of 262 K for the test set, metrics that we outperform with our models. Recently, Hong and co-workers have published multiple versions of deep learning models^{43,64} based on a graph neural network for materials embedding from compositions only and a residual neural network for melting temperature predictions. Their models are freely accessible and can be used to predict melting temperatures of compounds containing up to four elements with version 1⁹⁷ and also support predictions from an ensemble of 30 deep learning models with version 2 using the Materials Properties Prediction Application Programming Interface (API).^{98,99} Unlike our dataset, Hong's data contain entries from experiments and DFT-based simulations. Hong reports an R^2 training score of 0.99, an R^2 test score of 0.96, a training RMSE of 75 K, and a test RMSE of 138 K for their model for a dataset containing melting temperatures of more than 9300 materials. With version 1 of Hong's model, the following metrics were obtained: for our training set—RMSE: 304 K, MAE: 200 K, MAPE: 20%, R^2 : 0.85, and for our test set—RMSE: 404 K, MAE: 265 K, MAPE: 31%, R^2 : 0.64. The performance metrics for version 2 of Hong's model are for our training set – RMSE: 193 K, MAE:

109 K, MAPE: 10%, R^2 : 0.94, and for our test set – RMSE: 168 K, MAE: 116 K, MAPE: 11%, R^2 : 0.94. Version 2 also reports standard deviations for the predicted melting temperatures of 63 and 78 K for the training and test sets, respectively. We note that our dataset might contain materials that Hong's model was trained on (that might even have different melting temperature values tabulated from our dataset), and we have made no effort to correct this bias. Hong's model has overall lower reported error metrics and is generalizable and transferable to larger compositional materials spaces. While it is not an entirely fair comparison, since Hong's model is suitable for a broader range of materials and does not require features from DFT calculations, it is encouraging that our simple and computationally extremely efficient tree ensemble model can deliver state-of-the-art predictive performance on a similar order of magnitude as significantly more complex ensembles of deep learning models.

V. CONCLUSION

We evaluated the predictive performance of different machine learning methods for melting temperature prediction of binary inorganic solid materials. We first considered a direct supervised-learning approach, finding that SHAP analysis identified the cohesive energy and elastic moduli as the most important features. We found that the fidelity of predictions can further be improved by introducing an additional unsupervised-learning step that first classifies the materials based on their bonding characteristics before melting-point regression. Not only does this two-step model exhibit improved accuracy, but the approach also provides additional insights into feature importance and different types of melting that depend on the degree of ionicity in the bonding inside a material. Finally, we employed symbolic learning to establish interpretable physical models for the melting temperature that highlighted the significance of the top-performing features from the previous approaches and gave more insights into the theory behind melting and led to a simple linear equation for estimating the melting temperature of non-oxides and non-halides.

SUPPLEMENTARY MATERIAL

The [supplementary material](#) comprises 15 supplementary figures and two supplementary tables with error metrics.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (NSF) under Grant No. DMR1940290 (Harnessing the Data Revolution, HDR). We acknowledge computing resources from Columbia University's Shared Research Computing Facility project, which is supported by NIH Research Facility Improvement under Grant No. 1G20RR030893-01, and associated funds from the New York State Empire State Development, Division of Science Technology and Innovation (NYSTAR), under Contract No. C090171, both awarded April 15, 2010. This work also made use of the Illinois Campus Cluster, a computing resource that is operated by the Illinois Campus Cluster Program (ICCP) in conjunction with the National Center for Supercomputing Applications (NCSA) and which is supported

by funds from the University of Illinois Urbana-Champaign. The research of SC was partially supported by the National Science Foundation (NSF) under Grant Nos. 1939916 and 1939956.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Vahe Gharakhanyan: Data curation (equal); Investigation (equal); Methodology (equal); Software (lead); Writing – original draft (lead); Writing – review & editing (equal). **Luke J. Wirth:** Investigation (equal); Methodology (equal); Writing – original draft (supporting); Writing – review & editing (equal). **Jose A. Garrido Torres:** Data curation (supporting); Writing – review & editing (equal). **Ethan Eisenberg:** Data curation (equal); Writing – review & editing (equal). **Ting Wang:** Data curation (equal); Writing – review & editing (equal). **Dallas R. Trinkle:** Conceptualization (supporting); Funding acquisition (equal); Supervision (supporting); Writing – review & editing (equal). **Snigdhasu Chatterjee:** Conceptualization (supporting); Funding acquisition (equal); Supervision (supporting); Writing – review & editing (equal). **Alexander Urban:** Conceptualization (lead); Funding acquisition (equal); Methodology (equal); Supervision (lead); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

All experimental reference data, material fingerprints, and trained models have been posted on GitHub and are freely accessible at <https://github.com/atomisticnet/MeLting>.

The ML methodology of the present work is implemented in Python. The provided code yields vectorized compound fingerprints and can generate predictive models based on a selected regression model, perform hyperparameter tuning, as implemented in *scikit-learn*, validate, and test the models. MeLting also uses the Python Materials Genomics (*pymatgen*) package¹⁰⁰ for materials fingerprint construction. The source code for this work, including an example Python notebook, has been published on GitHub at <https://github.com/atomisticnet/MeLting> and is freely accessible under the terms of the MIT license.

REFERENCES

- ¹D. R. Sadoway, “New opportunities for metals extraction and waste treatment by electrochemical processing in molten salts,” *J. Mater. Res.* **10**, 487–492 (1995).
- ²A. Allamore, “Features and challenges of molten oxide electrolytes for metal extraction,” *J. Electrochem. Soc.* **162**, E13–E22 (2015).
- ³A. Allamore, “Electrochemical engineering for commodity metals extraction,” *Electrochem. Soc. Interface* **26**, 63 (2017).
- ⁴B. R. Nakanishi and A. Allamore, “Electrochemical investigation of molten lanthanum-yttrium oxide for selective liquid rare-earth metal extraction,” *J. Electrochem. Soc.* **166**, E420 (2019).
- ⁵K. Daehn and A. Allamore, “Electrolytic production of copper from chalcopyrite,” *Curr. Opin. Electrochem.* **22**, 110 (2020).
- ⁶A. Allamore and L. Rush, “High temperature electrolysis: Combining metal extraction and energy storage?,” *ECS Meet. Abstr.* **MA2020-01**, 1177 (2020).
- ⁷*Sustainable Metals Management*, edited by A. Gleich, R. U. Ayres, and S. Gößling-Reisemann (Springer, Dordrecht, 2006), Vol. 19.
- ⁸J. Xu *et al.*, “A review of processes and technologies for the recycling of lithium-ion secondary batteries,” *J. Power Sources* **177**, 512–527 (2008).
- ⁹K. Lu, “The future of metals,” *Science* **328**, 319–320 (2010).
- ¹⁰T. E. Graedel *et al.*, “What do we know about metal recycling rates?,” *J. Ind. Ecol.* **15**, 355–366 (2011).
- ¹¹H. Kim, J. Paramore, A. Allamore, and D. R. Sadoway, “Electrolysis of molten iron oxide with an iridium anode: The role of electrolyte basicity,” *J. Electrochem. Soc.* **158**, E101 (2011).
- ¹²B. K. Reck and T. E. Graedel, “Challenges in metal recycling,” *Science* **337**, 690–695 (2012).
- ¹³S. K. Sahu, B. Chmielowiec, and A. Allamore, “Electrolytic extraction of copper, molybdenum and rhenium from molten sulfide electrolyte,” *Electrochim. Acta* **243**, 382–389 (2017).
- ¹⁴X. Zheng *et al.*, “A mini-review on metal recycling from spent lithium ion batteries,” *Engineering* **4**, 361–370 (2018).
- ¹⁵X. Xi, M. Feng, L. Zhang, and Z. Nie, “Applications of molten salt and progress of molten salt electrolysis in secondary metal resource recovery,” *Int. J. Miner. Metall. Mater.* **27**, 1599–1617 (2020).
- ¹⁶A. Yazawa, “Thermodynamic considerations of copper smelting,” *Can. Metall. Q.* **13**, 443–453 (1974).
- ¹⁷W. E. Haupin, “Electrochemistry of the Hall-Heroult process for aluminum smelting,” *J. Chem. Educ.* **60**, 279 (1983).
- ¹⁸T. Mäkinen and P. Taskinen, “State of the art in nickel smelting: Direct outokumpu nickel technology,” *Miner. Process. Extr. Metall.* **117**, 86–94 (2008).
- ¹⁹H. Kvande and P. A. Drablos, “The aluminum smelting process and innovative alternative technologies,” *J. Occup. Environ. Med.* **56**, S23 (2014).
- ²⁰F. A. Lindemann, “The calculation of molecular vibration frequencies,” *Phys. Z.* **11**, 609 (1910).
- ²¹M. Born, “Thermodynamics of crystals and melting,” *J. Chem. Phys.* **7**, 591–603 (1939).
- ²²Z. H. Jin, P. Gumbsch, K. Lu, and E. Ma, “Melting mechanisms at the limit of superheating,” *Phys. Rev. Lett.* **87**, 055703 (2001).
- ²³R. W. Cahn, “Melting from within,” *Nature* **413**, 582–583 (2001).
- ²⁴A. V. Granato, D. M. Joncich, and V. A. Khonik, “Melting, thermal expansion, and the Lindemann rule for elemental substances,” *Appl. Phys. Lett.* **97**, 171911 (2010).
- ²⁵F. Hummel, G. Kresse, J. C. Dyre, and U. R. Pedersen, “Hidden scale invariance of metals,” *Phys. Rev. B* **92**, 174116 (2015).
- ²⁶G. Kresse and J. Hafner, “*Ab initio* molecular dynamics for liquid metals,” *Phys. Rev. B* **47**, 558–561 (1993).
- ²⁷O. Sugino and R. Car, “*Ab Initio* molecular dynamics study of first-order phase transitions: Melting of silicon,” *Phys. Rev. Lett.* **74**, 1823–1826 (1995).
- ²⁸L. G. Wang and A. van de Walle, “*Ab initio* calculations of the melting temperatures of refractory bcc metals,” *Phys. Chem. Chem. Phys.* **14**, 1529–1534 (2012).
- ²⁹A. B. Belonoshko, S. Arapan, R. Martonak, and A. Rosengren, “MgO phase diagram from first principles in a wide pressure-temperature range,” *Phys. Rev. B* **81**, 054110 (2010).
- ³⁰A. B. Belonoshko and A. Rosengren, “High-pressure melting curve of platinum from *ab initio* Z method,” *Phys. Rev. B* **85**, 174104 (2012).
- ³¹D. Corradini, D. Marrocchelli, P. A. Madden, and M. Salanne, “The effect of dispersion interactions on the properties of LiF in condensed phases,” *J. Phys.: Condens. Matter* **26**, 244103 (2014).
- ³²A. Aguado and P. A. Madden, “New insights into the melting behavior of MgO from molecular dynamics simulations: The importance of premelting effects,” *Phys. Rev. Lett.* **94**, 068501 (2005).
- ³³D. Alfè, “Melting curve of MgO from first-principles simulations,” *Phys. Rev. Lett.* **94**, 235701 (2005).
- ³⁴J. H. Jung, P. Srinivasan, A. Forslund, and B. Grabowski, “High-accuracy thermodynamic properties to the melting point from *ab initio* calculations aided by machine-learning potentials,” *npj Comput. Mater.* **9**, 3–12 (2023).

- ³⁵L. Vočadlo and D. Alfè, “*Ab initio* melting curve of the fcc phase of aluminum,” *Phys. Rev. B* **65**, 214105 (2002).
- ³⁶S.-T. Lin, M. Blanco, and W. A. Goddard, “The two-phase model for calculating thermodynamic properties of liquids from molecular dynamics: Validation for the phase diagram of Lennard-Jones fluids,” *J. Chem. Phys.* **119**, 11792–11805 (2003).
- ³⁷R. S. DeFever, H. Wang, Y. Zhang, and E. J. Maginn, “Melting points of alkali chlorides evaluated for a polarizable and non-polarizable model,” *J. Chem. Phys.* **153**, 011101 (2020).
- ³⁸T. Porter, M. M. Vaka, P. Steenblik, and D. Della Corte, “Computational methods to simulate molten salt thermophysical properties,” *Commun. Chem.* **5**, 69 (2022).
- ³⁹L. Kaufman and H. Bernstein, *Computer Calculation of Phase Diagrams. With Special Reference to Refractory Metals* (Academic Press, 1970).
- ⁴⁰Z.-K. Liu, “Thermodynamics and its prediction and CALPHAD modeling: Review, state of the art, and perspectives,” *Calphad* **82**, 102580 (2023).
- ⁴¹A. Seko, T. Maekawa, K. Tsuda, and I. Tanaka, “Machine learning with systematic density-functional theory calculations: Application to melting temperatures of single- and binary-component solids,” *Phys. Rev. B* **89**, 054303 (2014).
- ⁴²P.-W. Guan and V. Viswanathan, “MeltNet: Predicting alloy melting temperature by machine learning,” *arXiv:2010.14048* (2020).
- ⁴³Q.-J. Hong, “A melting temperature database and a neural network model for melting temperature prediction,” *arXiv:2110.10748* (2021).
- ⁴⁴G. Pilania, J. E. Gubernatis, and T. Lookman, “Structure classification and melting temperature prediction in octet AB solids via machine learning,” *Phys. Rev. B* **91**, 214302 (2015).
- ⁴⁵J. Qin, Z. Liu, M. Ma, and Y. Li, “Machine learning-assisted materials design and discovery of low-melting-point inorganic oxides for low-temperature cofired ceramic applications,” *ACS Sustain. Chem. Eng.* **10**, 1554–1564 (2022).
- ⁴⁶Y. Saad *et al.*, “Data mining for materials: Computational experiments with AB compounds,” *Phys. Rev. B* **85**, 104104 (2012).
- ⁴⁷G. H. Wolf and R. Jeanloz, “Lindemann Melting Law: Anharmonic correction and test of its validity for minerals,” *J. Geophys. Res.: Solid Earth* **89**, 7821–7835 (1984).
- ⁴⁸X. H. Zheng and J. C. Earnshaw, “On the Lindemann criterion in 2D,” *Europhys. Lett.* **41**, 635 (1998).
- ⁴⁹M. M. Vopson, N. Rogers, and I. Hepburn, “The generalized Lindemann melting coefficient,” *Solid State Commun.* **318**, 113977 (2020).
- ⁵⁰M. Ross, “Generalized Lindemann melting law,” *Phys. Rev.* **184**, 233–242 (1969).
- ⁵¹S. A. Khrapak, “Lindemann melting criterion in two dimensions,” *Phys. Rev. Res.* **2**, 012040 (2020).
- ⁵²P. Hohenberg and W. Kohn, “Inhomogeneous electron gas,” *Phys. Rev.* **136**, B864–B871 (1964).
- ⁵³W. Kohn and L. J. Sham, “Self-consistent equations including exchange and correlation effects,” *Phys. Rev.* **140**, A1133–A1138 (1965).
- ⁵⁴Q.-J. Hong and A. van de Walle, “Solid-liquid coexistence in small systems: A statistical method to calculate melting temperatures,” *J. Chem. Phys.* **139**, 094114 (2013).
- ⁵⁵J. R. Morris, C. Z. Wang, K. M. Ho, and C. T. Chan, “Melting line of aluminum from simulations of coexisting phases,” *Phys. Rev. B* **49**, 3109–3115 (1994).
- ⁵⁶L.-F. Zhu, B. Grabowski, and J. Neugebauer, “Efficient approach to compute melting properties fully from *ab initio* with application to Cu,” *Phys. Rev. B* **96**, 224202 (2017).
- ⁵⁷Y. Liu *et al.*, “Melting point prediction of energetic materials via continuous heating simulation on solid-to-liquid phase transition,” *ACS Omega* **4**, 4320–4324 (2019).
- ⁵⁸J. Mei and J. W. Davenport, “Free-energy calculations and the melting point of Al,” *Phys. Rev. B* **46**, 21–25 (1992).
- ⁵⁹G. A. de Wijs, G. Kresse, and M. J. Gillan, “First-order phase transitions by first-principles free-energy calculations: The melting of Al,” *Phys. Rev. B* **57**, 8223–8234 (1998).
- ⁶⁰V. L. Deringer, M. A. Caro, and G. Csányi, “Machine learning interatomic potentials as emerging tools for materials science,” *Adv. Mater.* **31**, 1902765 (2019).
- ⁶¹J. Qi, A. M. Cheung, and S. J. Poon, “High entropy alloys mined from binary phase diagrams,” *Sci. Rep.* **9**, 15501 (2019).
- ⁶²S. L. Chen *et al.*, “On the calculation of multicomponent stable phase diagrams,” *J. Phase Equilib.* **22**, 373–378 (2001).
- ⁶³Q.-J. Hong, “Melting temperature prediction via first principles and deep learning,” *arXiv:2204.04740* (2022).
- ⁶⁴Q.-J. Hong, S. V. Ushakov, A. van de Walle, and A. Navrotsky, “Melting temperature prediction using a graph neural network model: From ancient minerals to new materials,” *Proc. Natl. Acad. Sci. U. S. A.* **119**, e2209630119 (2022).
- ⁶⁵Q.-J. Hong, “Melting temperature prediction via first principles and deep learning,” *Comput. Mater. Sci.* **214**, 111684 (2022).
- ⁶⁶R. Ouyang, S. Curtarolo, E. Ahmetcik, M. Scheffler, and L. M. Ghiringhelli, “SISSO: A compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates,” *Phys. Rev. Mater.* **2**, 083802 (2018).
- ⁶⁷Z. Guo, S. Hu, Z.-K. Han, and R. Ouyang, “Improving symbolic regression for predicting materials properties with iterative variable selection,” *J. Chem. Theory Comput.* **18**, 4945–4951 (2022).
- ⁶⁸D. L. Perry, *Handbook of Inorganic Compounds* (CRC Press, 2016).
- ⁶⁹M. L. Huber, A. H. Harvey, H. Horton, H. Ryffel, and C. McCauley, *CRC Handbook of Chemistry and Physics*, 92nd edition (CRC Press, Boca Raton, 2011).
- ⁷⁰M. de Jong *et al.*, “Charting the complete elastic properties of inorganic crystalline compounds,” *Sci. Data* **2**, 150009 (2015).
- ⁷¹Elastic Constants <https://docs.materialsproject.org/methodology/materials-methodology/elasticity>.
- ⁷²A. Jain *et al.*, “Commentary: The materials project: A materials genome approach to accelerating materials innovation,” *APL Mater.* **1**, 011002 (2013).
- ⁷³M. de Jong *et al.*, “A statistical learning framework for materials science: Application to elastic moduli of k-nary inorganic polycrystalline compounds,” *Sci. Rep.* **6**, 34256 (2016).
- ⁷⁴L. Pauling, “The sizes of ions and the structure of ionic crystals,” *J. Am. Chem. Soc.* **49**, 765–790 (1927).
- ⁷⁵L. Buitinck *et al.*, “API design for machine learning software: Experiences from the scikit-learn project,” *arXiv:1309.0238* (2013).
- ⁷⁶G. Ke *et al.*, “LightGBM: A highly efficient gradient boosting decision tree,” in *Advances in Neural Information Processing Systems* (Curran Associates, Inc., 2017), Vol. 30.
- ⁷⁷T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery And Data Mining* (ACM, San Francisco, CA, 2016), pp. 785–794.
- ⁷⁸S. M. LaValle, M. S. Branicky, and S. R. Lindemann, “On the relationship between classical grid search and probabilistic roadmaps,” *Int. J. Robot. Syst.* **23**, 673–692 (2004).
- ⁷⁹J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *J. Mach. Learn. Res.* **13**, 281–305 (2012).
- ⁸⁰S. Lloyd, “Least squares quantization in PCM,” *IEEE Trans. Inf. Theory* **28**, 129–137 (1982).
- ⁸¹P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *J. Comput. Appl. Math.* **20**, 53–65 (1987).
- ⁸²M. Kozak, “‘A dendrite method for cluster analysis’ by Caliński and Harabasz: A classical work that is far too often incorrectly cited,” *Commun. Stat. - Theory Methods* **41**, 2279–2280 (2012).
- ⁸³F. R. S. Karl Pearson, “LIII. On lines and planes of closest fit to systems of points in space,” *London Edinburgh Dublin Philos. Mag. J. Sci.* **2**, 559–572 (1901).
- ⁸⁴S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems* (Curran Associates, Inc., 2017), Vol. 30.
- ⁸⁵A game theoretic approach to explain the output of any machine learning model <https://github.com/shap/shap>.
- ⁸⁶S. I. Ranganathan and M. Ostojia-Starzewski, “Universal elastic anisotropy index,” *Phys. Rev. Lett.* **101**, 055504 (2008).
- ⁸⁷P. E. Blöchl, “Projector augmented-wave method,” *Phys. Rev. B* **50**, 17953–17979 (1994).

- ⁸⁸G. Kresse and J. Furthmüller, “Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set,” *Phys. Rev. B* **54**, 11169–11186 (1996).
- ⁸⁹G. Kresse and D. Joubert, “From ultrasoft pseudopotentials to the projector augmented-wave method,” *Phys. Rev. B* **59**, 1758–1775 (1999).
- ⁹⁰J. P. Perdew, K. Burke, and M. Ernzerhof, “Generalized gradient approximation made simple,” *Phys. Rev. Lett.* **77**, 3865–3868 (1996).
- ⁹¹D. H. Chung and W. R. Buessem, “The Voigt-Reuss-Hill approximation and elastic moduli of polycrystalline MgO, CaF₂, β -ZnS, ZnSe, and CdTe,” *J. Appl. Phys.* **38**, 2535–2540 (1967).
- ⁹²J. Rogers and S. Gunn, “Identifying feature relevance using a random forest,” in *Subspace, Latent Structure and Feature Selection* (Springer, Berlin, Heidelberg, 2006), pp. 173–184.
- ⁹³C. Greskovich and J. H. Rosolowski, “Sintering of covalent solids,” *J. Am. Ceram. Soc.* **59**, 336–343 (1976).
- ⁹⁴J. H. Williams, *Crystal Engineering: How Molecules Build Solids* (Morgan & Claypool Publishers, 2017).
- ⁹⁵B. Ouyang *et al.*, “Synthetic accessibility and stability rules of NASICONs,” *Nat. Commun.* **12**, 5752 (2021).
- ⁹⁶L. Wang and B. Ouyang, “Phase selection rules of multi-principal element alloys,” *Adv. Mater.* **36**, 2307860 (2024).
- ⁹⁷Melting Temperature Predictor Version 1 – Hong Research Group <https://faculty.engineering.asu.edu/hong/melting-temperature-predictor-version-1/>.
- ⁹⁸Melting Temperature Predictor Version 2 – Hong Research Group <https://faculty.engineering.asu.edu/hong/melting-temperature-predictor/>.
- ⁹⁹S.-D. Xue and Q.-J. Hong, “Materials Properties Prediction (MAPP): Empowering the prediction of material properties solely based on chemical formulas,” *arXiv:2311.05133* (2023).
- ¹⁰⁰S. P. Ong *et al.*, “Python materials genomics (pymatgen): A robust, open-source python library for materials analysis,” *Comput. Mater. Sci.* **68**, 314–319 (2013).