Journal of Applied Physics

RESEARCH ARTICLE | DECEMBER 11 2023

End-to-end material thermal conductivity prediction through machine learning

Yagyank Srivastava ; Ankit Jain

J. Appl. Phys. 134, 225101 (2023)
https://doi.org/10.1063/5.0183513

Journal of Applied Physics ARTICLE pubs.aip.org/aip/jap

End-to-end material thermal conductivity prediction through machine learning

Cite as: J. Appl. Phys. **134**, 225101 (2023); doi: 10.1063/5.0183513
Submitted: 23 October 2023 · Accepted: 18 November 2023 ·
Published Online: 11 December 2023

Yagyank Srivastava and Ankit Jain[a)]

AFFILIATIONS

Mechanical Engineering Department, IIT Bombay, India

Note: This paper is part of the special topic, Machine Learning for Thermal Transport.
[a)]**Author to whom correspondence should be addressed:** a_jain@iitb.ac.in

ABSTRACT

We investigated the accelerated prediction of the thermal conductivity of materials through end-to-end structure-based approaches employing machine learning methods. Due to the non-availability of high-quality thermal conductivity data, we first performed high-throughput calculations based on first principles and the Boltzmann transport equation for 225 materials, effectively more than doubling the size of the existing dataset. We assessed the performance of state-of-the-art machine learning models for thermal conductivity prediction on this expanded dataset and observed that all these models suffered from overfitting. To address this issue, we introduced a different graph-based neural network model, which demonstrated more consistent and regularized performance across all evaluated datasets. Nevertheless, the best mean absolute percentage error achieved on the test dataset remained in the range of 50–60%. This suggests that while these models are valuable for expediting material screening, their current accuracy is still limited.

I. INTRODUCTION

Thermal conductivity (κ) is an important material property critical in determining the performance and efficiency of devices in various technological applications such as thermoelectric energy generation, thermal insulation, and memory storage.[1–4] For many of these applications, low thermal conductivity semiconducting solids are desired, while for others (such as heat dissipation and microprocessors), materials with high κ are desired.[2,5,6] For materials used in most of these applications, thermal transport is dominated by atomic vibrations, i.e., phonons, with room temperature κ in the range of 0.1–3000 W/m K.[7] The traditional search for novel low and high κ materials is carried out experimentally using a trial-and-error approach. Lately, it has become possible to use *ab initio*-driven lattice dynamics calculations in conjunction with the Boltzmann transport equation to predict κ of materials.[6,8–10] Though these state-of-the-art quantum-mechanical calculations are instrumental in predicting correct κ without requiring any fitting parameter,[6,9,11,12] the computational cost of these calculations is very high at several hundred to several thousand cpu-hours for each material.[8] Thus, the application of such calculations is limited to simple material systems, and computational exploration of new materials is restricted to the simple substitution of one or two atomic species in known material systems.[13–16]

Recently, machine learning (ML) approaches have gained significance for data-driven exploration and discovery of materials. The ML models can be trained to learn the material properties/behavior from known/training datasets and the trained models can then be used to predict the properties/behavior of new material configurations. Such approaches have already been used successfully to predict the variety of material properties, for instance, ionic conductance,[17] crystal thermal conductivity,[18] thermoelectric figure of merit,[19] opto-electronic properties,[20,21] mechanical strength,[22] nuclear fuel systems,[23] and drug discovery.[24] For the particular case of thermal transport, while these approaches are gaining popularity, they are still limited.[25] For instance, Pal *et al.*[26] employed a scale-invariant ML model to accelerate the search of quaternary chalcogenides with low κ, Hu *et al.*[27] employed ML to minimize coherent heat conduction across aperiodic superlattices, Rodriguez *et al.*[28] trained neural network based interatomic forcefield to do bottom-up prediction of κ based on intermediate phonon properties such as mean square displacements and bonding/anti-bonding characters, Wan *et al.*[29] employed Bayesian optimization to optimize the thermal conductance of graphene nanoribbons, and Visaria and Jain[30] employed neural network based auto-encoders to do space transformation to search for material configurations

with low- and high-κ from the exponentially large search space of considered superlattices.

For direct end-to-end prediction of κ from material crystal structures, the notable contribution is by Zhu *et al.*[31] where $\log(\kappa)$ of diverse material systems is predicted using graph neural networks and random forest models and coefficient of determination (R^2) of 0.85 and 0.87 are obtained from the two models in a fivefold validation on the dataset consisting of 132 materials. Subsequently, Liu *et al.*[32] improved further on this and proposed a transfer learning approach to train a feedforward neural network and obtained an improved R^2 of 0.83 (compared to 0.67 with direct learning) on 170 materials in a fivefold cross-validation.

In both studies, however, the obtained performances are reported for fivefold cross-validation and the performances of these models on a different, completely unseen dataset, are not established. This was primarily done due to the limitation on the available datasets with only ~100 entries. Nonetheless, the evaluation of ML performance on unseen datasets is critical as (i) these employed models often have several thousand trainable parameters and as such, are severely prone to over-fitting for datasets of only ~ 100 datapoints, (ii) while optimizing the model performance, the test dataset is indirectly passed on to ML models via hyperparameter tuning, and as such, the performance of model on true unseen real-world-like data (completely unseen by the ML model) can be different than that on the test data.

With these in mind, in this work, we first carry out a high-throughput, *ab initio*-driven κ calculations to augment the currently available high-quality κ-dataset by more than twofold from the current size of 166 to 398 and, next, we develop a graph-based ML model to reduce the problem of over-fitting in κ prediction of materials.

II. METHODOLOGY

A. High-throughput κ calculations

While the full details regarding the calculation of thermal conductivity using the Boltzmann transport equation approach can be found elsewhere,[10,12] the thermal conductivity, κ_α, is obtained as[7,33,34]

$$\kappa = \sum_i c_{ph,i} v_\alpha^2 \tau_i, \tag{1}$$

where the summation is over all the phonon modes in the Brillouin zone enumerated by $i \equiv (q, \nu)$, where q and ν are phonon wavevector and mode index, and $c_{ph,i}$, v_α, and τ_i represent phonon specific heat, group velocity (α-component), and transport lifetime, respectively. The transport lifetimes are obtained by considering phonon–phonon scattering via three-phonon scattering processes.

The harmonic and anharmonic force constants that are required as an input to compute phonon dispersion and phonon scattering rates are obtained from density functional perturbation theory (DFPT) and density functional theory (DFT) calculations as implemented in the openly available quantum mechanical simulation package Quantum Espresso.[35,36] The plane wave-based basis set with norm-conserving Vanderbilt pseudopotentials[37] and the plane wave kinetic energy cutoff is set at 80 Ry in all calculations. The structure relaxations are performed using primitive unit cells and the electronic Brillouin zone is sampled using a Monkhorst–Pack wavevector grid of size k_i such that $k_i.|a_i^{pri}| \sim 30$Å, where $|a_i^{pri}|$ represents the length of primitive unit cell lattice vector $\mathbf{a}_i^{pri}$. The electronic total energy is converged to within 10^{-10} Ry/atom during self-consistent cycles and the structure relaxations are performed with a force convergence criterion of 10^{-5} Ry/Å.

The harmonic force constants and Born effective charges are obtained using DFPT calculations on primitive unit cells. The DFPT calculations are initially performed on phonon wavevector grids of size q_i^c such that $q_i^c.|a_i^{pri}| \sim 30$ Å and are later interpolated to grids of size q_i^f such that $q_i^f.|a_i^{pri}| \sim 100$ Å for three-phonon scattering calculations. The anharmonic force constants are obtained from Taylor-series fitting of Hellmann–Feynman forces obtained on 200 thermally populated supercells (corresponding to a temperature of 300 K) obtained from N_i repetitions of the conventional unit cell such that $N_i.|a_i^{conv}| \sim 15$ Å. The cubic force constant interaction cutoff is set at 6.5 Å for the majority of compounds, though for some compounds with lower symmetries, this value is reduced to 5.0 Å.

The material screening process for selecting compounds on which κ calculations are carried out is detailed in Fig. 1. Starting

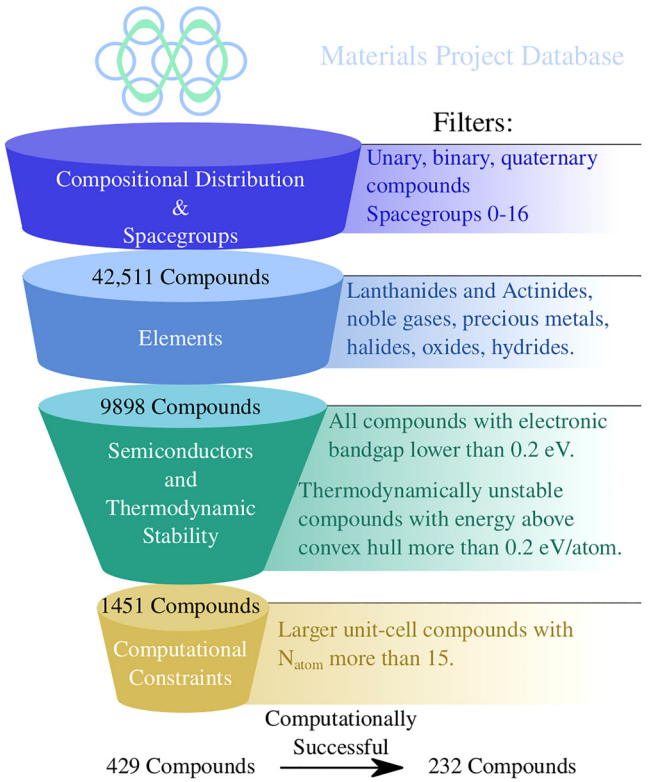

FIG. 1. The filtering criterion employed to select a pool of stable ternary semiconductors from the Materials Project database[38] for *ab initio*-driven high-throughput computation of phonon thermal conductivity.

with all ternary compounds from the Materials Project[38] within the orthorhombic, tetragonal, trigonal, hexagonal, and cubic space-groups, resulting in a total of 42 511 compounds, compounds containing lanthanides and actinides, noble gases, and precious metals (Au, Pt, Pd, Ir, Ru, Re, Rh, Hg, Hf) are removed, and the maximum atomic number of participating species is restricted to 83. Further, strongly ionic compounds formed by halides, oxides, and hydrides are also removed. Since the focus here is on the phonon thermal transport in semiconductors and insulators, compounds with electronic bandgap lower than 0.2 eV (as obtained from GGA-based DFT in the Materials Project) are also removed. The materials are further filtered on the basis of thermodynamic stability to have energy above the convex hull (with respect to all reported materials in the Materials Project) less than 0.2 eV/atom. Finally, considering the N_{atom}^4 scaling of computational cost for thermal conductivity calculations,[10] compounds with more than 15 atoms in the unit cell are also removed. Of the total of 429 filtered compounds, during structure relaxation and dynamical stability calculations, 197 compounds are further dropped due to the presence of imaginary phonon modes and/or non-convergence of self-consistent functional calculations, and the full thermal conductivity calculations are carried out for 232 compounds.

B. Machine learning models

1. Feed-forward neural network

For the feed-forward neural network (NN), we employed three fully connected layers with a random number of neurons in the first two layers (between 60 and 90 in the first layer and between 20 and 50 in the second layer), and 10 neurons in the final layer with 40 different seedings for initial weights. We generated 360 such models with different combinations of neurons and seedings. We employed Rectified Linear Unit (ReLU) as the activation/non-linear function and used Adam optimizer[39] with a learning rate of 0.001 and a batch size of 85. The training is done on 500 epochs with mean absolute error (MAE) loss. For direct feed-forward NN, the weights were randomly initialized from a standard normal distribution while for transfer-learning feed-forward NN, the weights are initialized by pre-training the network on a low-fidelity dataset (discussed in Sec. III). This selection of network architecture is similar to that employed in Ref. 32

2. Graph-based neural networks

For graph-based neural networks, we employed three different implementations: (a) crystal graph convolution neural network (CGCNN),[40] (b) materials graph network (MEGNet),[41] and (c) our own implementation (referred to as GNN). All three of these models are graph-based convolution neural networks and they differ in the implementation of node, edge, and/or state features. For instance, while CGCNN and GNN only have node and edge features, MEGNet also have state features. Similarly, while CGCNN employs one-hot encoding (of length 9) based on interatomic distance for edge features, MEGNet and GNN use only one attribute corresponding to actual value of interatomic spatial distance for edge features. The CGCNN and MEGNet are employed with default settings with MAE loss and further details on their implementation can be found in Refs. 40 and 41.

In GNN, each node represents an atomic site (i) in the unit cell with atomic number and position (reduced coordinates) as the node features. The edges represent atomic neighbors within 5 Å distance. The atomic numbers are embedded to a vector of length 10 using pytorch embedding[42] which is passed through a non-linear layer to obtain an initial node representation. The node position (X_i) and neighbor position (X_j) give the neighbor's distance vector ($X_j - X_i$), which is passed through a non-linear layer followed by a mean pooling to obtain a consolidated neighbor vector of node (i). Finally, node and neighbor vectors are concatenated and passed through a non-linear layer to obtain an updated node representation. This process of obtaining node embedding is repeated three times, and in the third iteration, the lattice parameters ($\alpha, \beta, \gamma, a, b, c$) are concatenated with the consolidated neighbor vector to obtain a final node representation. The final node embeddings are passed through mean and variance pooling followed by a fully connected layer to obtain a prediction for κ. The GNN model is trained with ReLU activation and MAE loss using an Adam optimizer with a learning rate of 0.0001.

3. Random forest and gaussian process

For random forest (RF) and gaussian process (GP), we employed the implementation of Sklearn.[43] For RF, we employed 50 tress and 40 distinct random states. For GP, we employed a composite kernel by adding three kernels (Radial Basis Function kernel,[44,45] Constant Kernel,[44,45] and White Kernel[44,46,47]) and optimized kernel hyper-parameters with the COBYLA method as implemented in Sklearn.[43] In contrast to other ML methods, GP also provides confidence on predicted values, which is useful in establishing the uncertainty of predicted κ.

C. Material fingerprints

For NN, RF, and GP, we generated a material fingerprint using elemental and compound features. For elemental features, we computed five statistical properties (mean, maximum, minimum, etc.) of 12 elemental properties (atomic number, electronegativity, atomic weight, etc.) of each participating element by employing the Matminer[48] ElementProperty interface with "magpie" data source to obtain elemental property vector of length 60. This elemental property vector is concatenated with three compound properties: space group, volume, density, and two Matminer's[48] IonProperty (mean and maximum of ionic character) to obtain a final material fingerprint of length 65. For graph-based models, the crystal structures are passed directly to the models, and the material fingerprints are determined internally by the models.

III. RESULTS

A. Thermal conductivity datasets

We considered three κ-datasets: (i) low-fidelity (LF) dataset consisting of 1507 materials with thermal conductivity obtained using a semi-empirical model,[32] (ii) high-fidelity (HF) dataset consisting of 166 materials whose thermal conductivities are reported in the literature from either experimental measurements or from

first-principle calculations,[32] and (iii) high-throughput (HT) dataset consisting of 232 ternary materials whose thermal conductivities are obtained using the full *ab initio* calculations with the same computational setting for all materials as detailed in Sec. II A (see Fig. 2). The data contained in LF and HF datasets are isotropic, while the data in HT dataset are direction resolved. As such, to stay consistent with other datasets, we performed directional averaging of κ for HT dataset and report all results for direction-averaged mean κ. The LF and HT datasets have similar κ distribution and have most materials with κ between 1 and 20 W/m K. In comparison, the HF dataset is centered between 10 and 100 W/m K and have larger number of compounds with high κ. It is worthwhile to emphasize here that even though the size of employed datasets in this study is $\sim$ 100, these are among the largest high-quality datasets available for κ. On top of a large 3–4 orders of magnitude span of material κ values, this scarcity of data also makes ML for κ more challenging compared to other material properties.

B. Evaluation metrices

The performance of ML models is generally characterized on hold-out test datasets consisting of 10%–30% of the total data, using metrics such as mean absolute error (MAE), mean square error (MSE), root mean square error (RMSE), coefficient of determination (R^2), and mean absolute percentage error (MAPE). For properties such as formation energy, electronic bandgap, mixing entropy, etc., the entire range of predicted property is less than an order of magnitude and all of these performance metrices are adequate. For κ, the range is over four orders of magnitude and, as such, the choice of performance metric is not obvious.

To illustrate this on real κ-dataset, we plot the model performances of four hypothetical models on LF dataset in Fig. 3. In Fig. 3(a), we considered a model that is able to perfectly predict all high-κ points and results in a random prediction for low-κ materials and in Fig. 3(b), we have a model with perfect prediction for low-κ materials and a random prediction for high-κ materials. For Fig. 3(a), even though the predictions are random for all low-κ materials, the obtained R^2 value is high at 0.97, while in Fig. 3(b), when randomization is applied to high-κ points instead, the R^2 value is negative; thus demonstrating the bias of R^2 toward large value datapoints. Similarly, in Figs. 3(c) and 3(d), the considered models predict random values for low-κ datapoints but they differ in that while model in Fig. 3(c) under-predicts, the model in Fig. 3(d) over-predicts these low-κ datapoints. Since the performance is perfect for all high-κ datapoints, the obtained R^2 is close to unity in both cases. However, for MAPE, while the obtained value is low at only 10% for Fig. 3(c), the value is much higher (93%) for Fig. 3(d); thus, indicating the bias of MAPE to underpredict datapoints.

To account for this challenge arising from κ spanning over multiple orders of magnitude, the model performance for κ prediction is often evaluated on a log-scale where log-κ varies in the range ~ -1 to 3. However, this scale is also tricky as the predictions that look good on log-scale are actually severely off due to the exponential nature of this scale. For instance, the predicted log-κ for diamond of 3.7 compared to the experimentally measured value of 3.4 is off by 100% even though the difference is less than 10% on the log-scale.

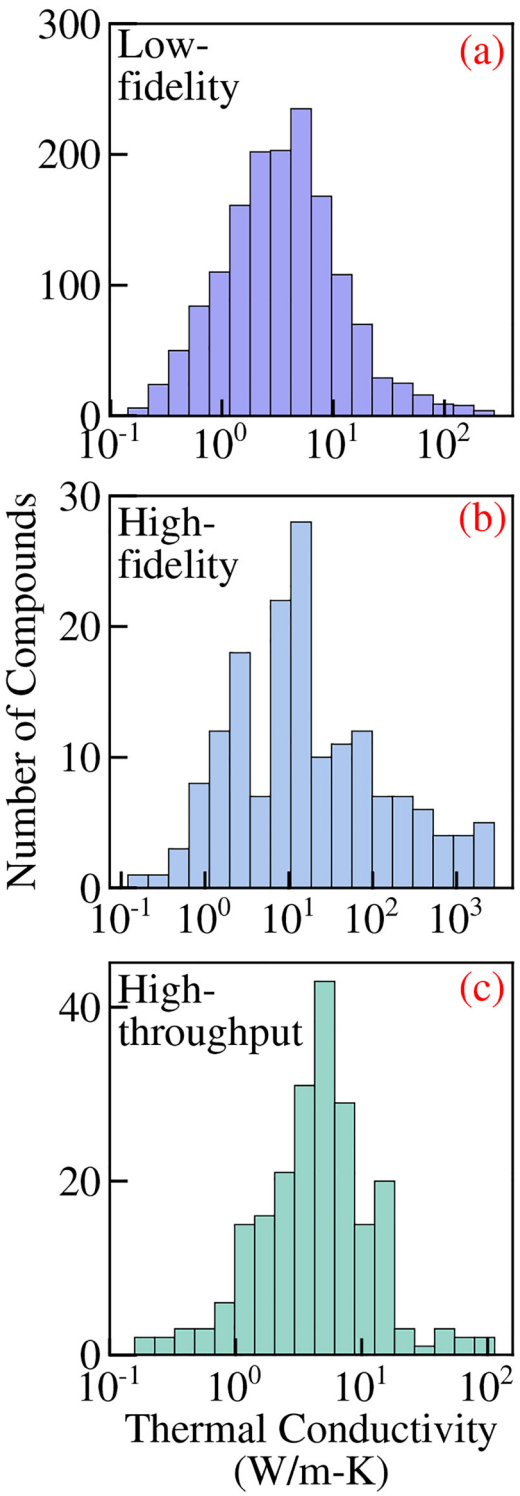

FIG. 2. The number distribution of κ in (a) low-fidelity,[32] (b) high-fidelity,[32] and (c) high-throughput datasets.

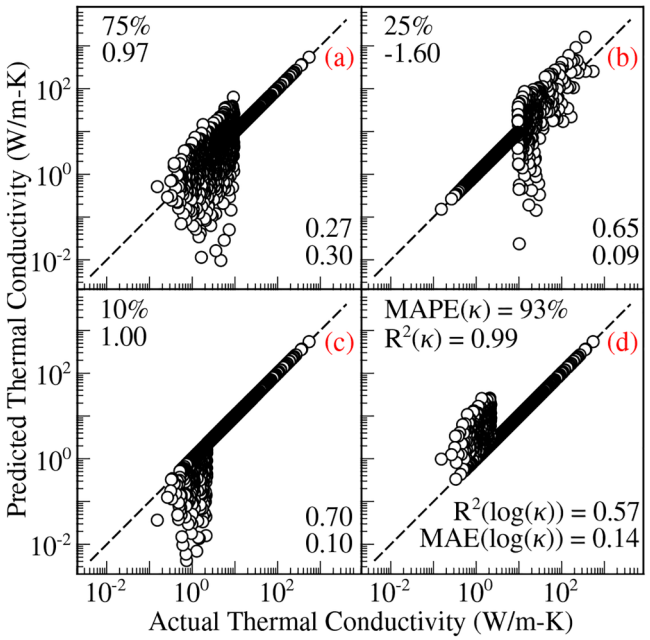

FIG. 3. The variation of various performance metrics when model predictions are: (a) perfect on all high κ ($\kappa > 10$ W/m K) and random on low κ ($\kappa \leq 10$ W/m K), (b) perfect on all low κ and random on high κ, (c) perfect on all high κ and underprediction on low κ, and (d) perfect on all high κ and overprediction on low κ. The considered datapoints are from LF dataset having the actual distribution of κ values. The $R^2(\kappa)$ is biased toward high-value datapoints and MAPE(κ) is biased toward overprediction of actual values.

In this work, we report MAE on log κ [MAE(log(κ))], MAPE, and R^2 [on κ and log(κ)] for all models and we compare model performances based on MAE(log(κ)).

C. Thermal conductivity prediction

We start by first testing the performance of the trained NN model on the new HT dataset. For this, we pre-train the NN model on the LF dataset and then employ transfer learning and train the model on the HF dataset. We find the model performance in five-fold cross-validation on the HF dataset is 0.86 [$R^2(\kappa)$], which is similar to that reported by Liu *et al.*[32] [$R^2(\kappa)$ of 0.83]. However, when this trained NN model is used to predict κ of materials from the HT dataset, the obtained prediction performance is unsatisfactory with MAPE(κ) of 311%, $R^2(\log\kappa)$ of −0.41, $R^2(\kappa)$ of −8.00, and MAE($\log\kappa$) of 0.44. As discussed above, this is primarily owing to the small dataset size of the HF dataset. As such, the HF and HT datasets are combined in this study and model performances are evaluated on this combined HF/HT dataset.

We train the considered ML models on the LF dataset and evaluate their performance for the prediction of κ on HF/HT datasets (Fig. 4). We find that all ML models can fit the κ variation in the LF dataset correctly with test MAE(log(κ)) of 0.16 or less. The best train performance is by GPR and NN models, each with test MAE(log(κ)) less than 0.14. As emphasized above in Sec. II B, we find that, even though all ML models have test MAE(log(κ)) between 0.14–0.16, the R^2 varied between 0.57 and 0.83 depending on the quality of fit of high κ datapoints, thus further highlighting the limitations of R^2 as the performance metric. Note that this limitation is an outcome of orders of magnitude scale of κ and when R^2 is evaluated on log κ, the obtained value is between 0.81 and 0.88 for all considered models.

When we use these trained models to predict κ of materials from HF/HT datasets, we find that the MAE(log(κ)) of all models is high at > 0.38. Nevertheless, considering that the LF dataset is derived based on empirical relations and is not a true representative of DFT result/experimental measurements (data in HF/HT datasets), it is impressive to see that all considered models can capture the overall general trend of κ. The best performance is by the GNN model with MAE(log(κ)) of 0.38. This same model also resulted in the best generalization between the two datasets as can be estimated based on the ratio of prediction MAE(log(κ)) on HF/HT datasets to test MAE(log(κ)) on the LF dataset. Noticeably, this ratio is only 2.35 compared to more than 2.6 for all other models. Furthermore, the GPR and NN models, which resulted in the best performances in testing on the LF dataset, delivered the worst generalization with prediction MAE(log(κ)) of 0.43 and 0.41 on the HF/HT dataset; thus suggesting potential over-fitting by these models compared to learning of general trends by the GNN model. It is worthwhile to note here that the number of trainable weights in GNN and NN are similar (between 7000 and 10000), and, as such, the regularized performance by GNN is due to its specific architecture as presented in Sec. II B.

Next, we train the considered ML models directly on HF/HT datasets (total of 398 datapoints) and evaluate model performances on the hold-out dataset (Fig. 5). We generated the train-test split by sorting the materials by their κ and by placing every fifth entry from this sorted list in the test dataset (80:20 split) for all models. During training, we find that the MAE(log(κ)) for RF, GPR, NN, and MegNet models reduced compared to their corresponding test values for the LF dataset. For instance, for NN, the MAE(log(κ)) reduced from 0.14 for testing on the LF dataset to 0.04 for training on HF/HT datasets. This reduction is understandable as the number of datapoints used in HF/HT training is only 398 compared to 1507 on the LF dataset. Surprisingly, for GNN and CGCNN, this is not true, and the MAE(log(κ)) is higher for training on HF/HT datasets than for testing on the LF dataset. Both GNN and CGCNN resulted in MAE(log(κ)) higher than 0.21 (0.21 and 0.25, respectively) during training on the HF/HT compared to less than 0.10 by all other models. This large MAE(log(κ)) from GNN and CGCNN seems to suggest the inferior performance of these models compared to other models, but when we employed the trained models to predict κ of hold-out test dataset (75 datapoints), we found all models resulted in test MAE(log(κ)) > 0.25, thus suggesting over-fitting by all models. Compared to other models, for GNN and CGCNN, the over-fitting is less pronounced. For GNN and CGCNN, the test errors are only 36% and 46% larger than the corresponding train errors, but for RF, GPR, NN, and MeGNet, the test errors are 161%, 571%, 166%, and 1411% larger than the train errors. For NN, we also included regularization via the Dropout method[49] (after first hidden layer) with 20%

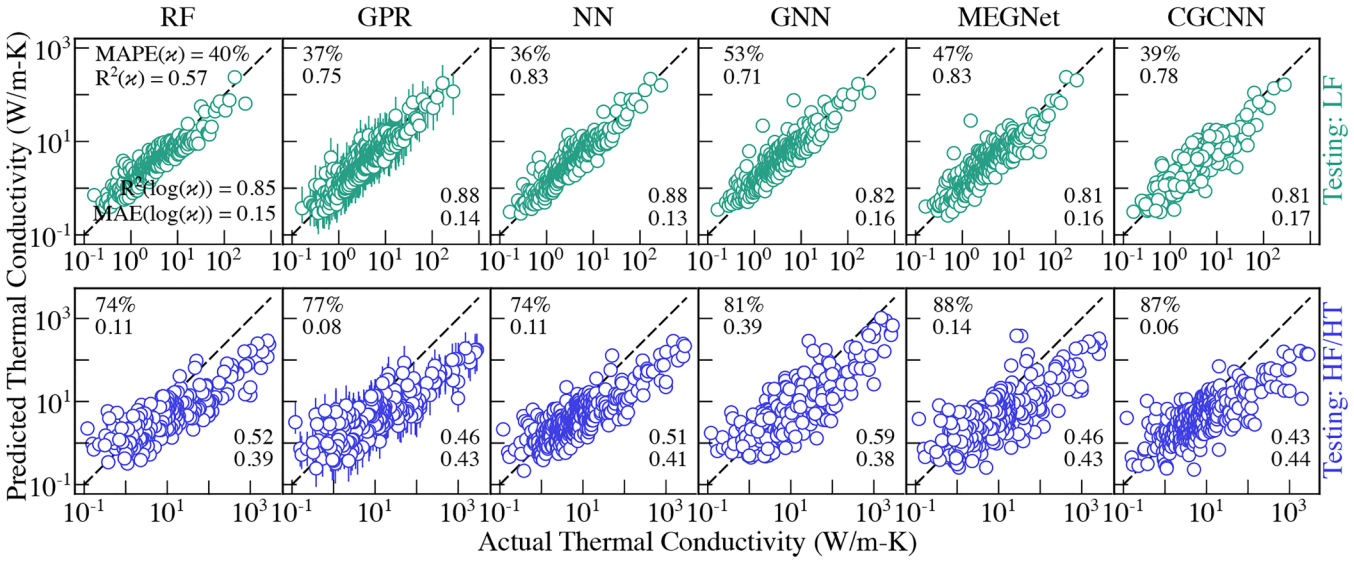

FIG. 4. The prediction performance of considered ML models on: (top panels) 20% of holdout compounds from the LF dataset and (bottom panels) HF/HT datasets. All models are trained on 80% of the LF dataset consisting of the same set of compounds. Without seeing any material from the HF/HT datasets, while all ML models are able to capture the general κ trend of these datasets (bottom panels), the best performance is obtained from the GNN model.

and 50% dropout rates, but in both cases, we found that the obtained test error was $> 160\%$ larger than the training error.

Compared to other models, in general, we find that the performance of graph-based GNN and CGCNN is regularized for end-to-end κ prediction. For GNN, we find that the performance is better than CGCNN. We believe that this is due to two major architectural differences between GNN and CGCNN: (i) the graph edges are represented using the actual neighbor distances in GNN as opposed to one-hot encoded vectors in CGCNN and (ii) the pooling employed in GNN preserves edge features to a larger extent by employing both mean as well as variance pooling as opposed to only mean pooling in CGCNN.

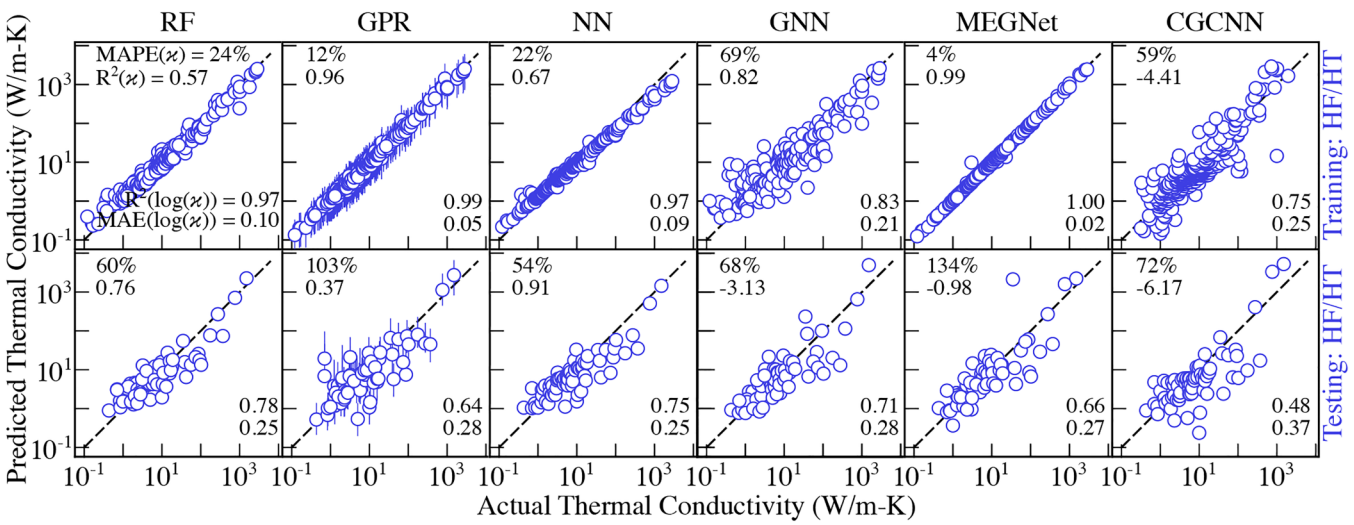

FIG. 5. The training (top panels) and testing (bottom panels) performance of considered ML models on HF/HT datasets. All models are trained on 80% of the HF/HT datasets consisting of the same set of compounds. GNN and CGCNN deliver the best-regularized performances, while all other models suffer from over-fitting.

Motivated by this impressive performance of GNN, along with its associated regularization, we explore the effect of transfer learning on GNN by pre-training it first on the LF dataset, followed by training on the HF/HT datasets (Fig. 6). For reference, we also included the NN model. As suggested by Liu *et al.*[32] for NN, we find that the prediction performance of GNN also improves with transfer learning. The test MAE(log(κ)) from GNN is reduced to 0.25 with transfer learning, which is similar to that from the NN model.

For materials exploration, it is imperative to have a confidence interval on the predicted value of κ from ML models. For the GPR model, the confidence intervals are analytically obtained from the model itself depending on the kernel similarity between train and test datapoints. However, as shown earlier, for κ prediction, GPR results in a severe over-fitting and the predicted κ from GPR model is out of one standard deviation bounds for a large number of test datapoints in Fig. 5. As an alternative, we pick the predicted κ variation between different ML models to measure confidence bounds.

In Fig. 7, the predicted κ for test materials from HF/HT datasets is re-plotted by combining predictions from RF, NN, GNN, and CGCNN models (GPR and MEGNet are omitted as both of these models result in severe over-fitting with test error being more than 500% larger than the train error). The lower/upper bounds are obtained by taking the minimum/maximum over all models, and the predicted value is obtained by taking the average of all models. Further, the materials for which true κ falls outside the lower/upper bounds are highlighted using empty markers in Fig. 7.

With this combined model, we find that the prediction performance is further improved and MAE(log(κ)) is reduced to 0.23. Further, the actual κ falls within this combined model's lower/upper bounds for around 50% of the 75 materials tested in Fig. 7.

Finally, we also tested the performance of trained ML models on some of the recently reported materials from the literature[50–59] after ensuring that these materials are not a part of any of the databases employed in model training/testing. We report these performances in Fig. 8. The predictions that fall within 50% of the literature-reported values are indicated using filled circles, while those falling outside of 50% are indicated using open circles in Fig. 8. We find that for extremely low κ, while predicted trends are right, all model predictions are off by more than 50% from the literature-reported values. This is, however, understandable as the lowest κ in our training dataset is 0.47 W/m K. Surprisingly, the NN model seems to predict values between 1 and 2 W/m K for all materials and results in a severe under-prediction for intermediate κ materials (with $\kappa > 4$ W/m K in Fig. 8). For materials with $\kappa > 0.47$ W/m-K, the GNN predictions are off by more than 50% only for two materials as compared to five by RF and CGCNN; thus, further highlighting the superior performance of GNN compared to all other ML models employed in this work.

IV. DISCUSSION

The application of ML methods for thermal transport can be broadly classified into three categories. In the first kind of application of ML for thermal properties, instead of direct prediction of κ/thermal properties, the interatomic forcefields are trained on DFT forces/energies and these trained forcefields are used in conjunction with approaches such as molecular dynamics simulations, lattice dynamics calculations, etc. to predict the thermal properties. These DFT-trained ML forcefields offer the advantage of capturing the right thermal transport physics (depending on the approach) and other than the employed approach, the accuracy of results depends entirely on the accuracy of the trained forcefield (which can now approach the same accuracy as that in DFT[60]). Further,

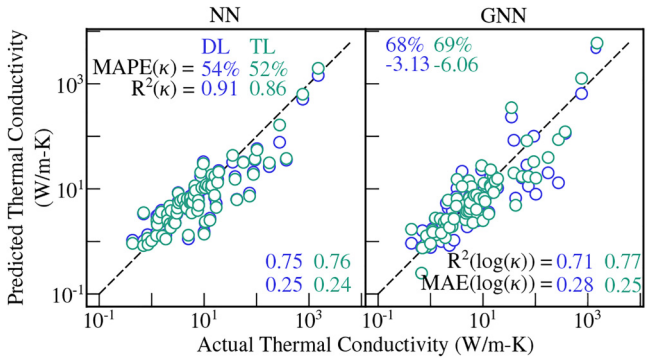

FIG. 6. The effect of transfer learning (TL) on the prediction performance of (left) NN model and (right) GNN model. In direct learning (DL), the models are directly trained on the HF/HT datasets while in TL, the models are first pre-trained on the LF dataset and are later re-trained on the HF/HT datasets. The reported performances are on the 20% of the hold-out dataset.

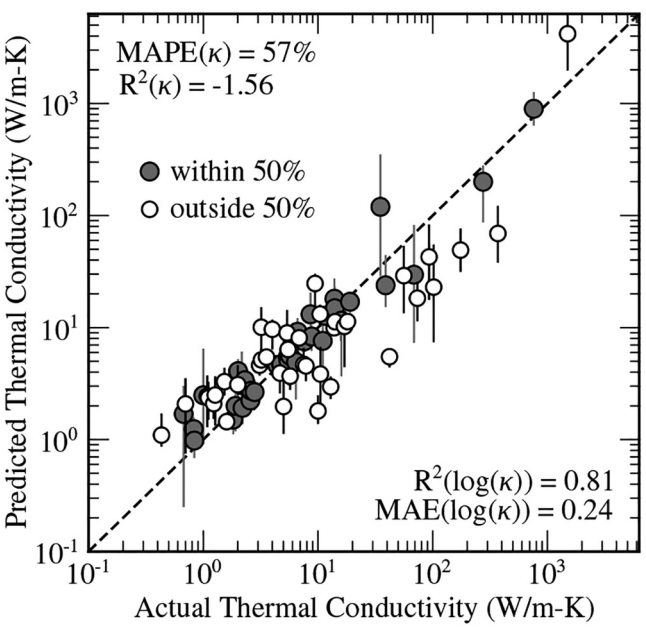

FIG. 7. The prediction performance of the combined model on 20% of holdout HF/HT datasets. The predicted values are taken as mean of RF, NN, GNN, and CGCNN models and the plotted lower/upper bounds are minimum/maximum predictions from these models.

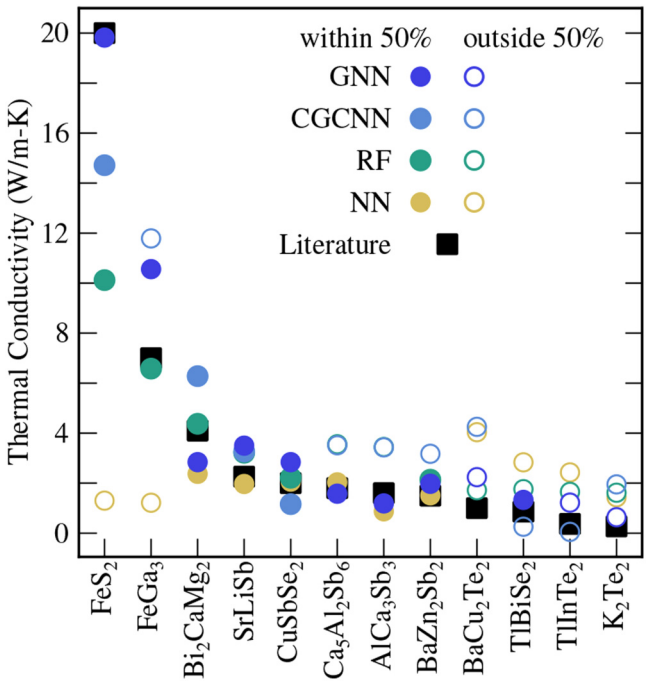

FIG. 8. The prediction performance of considered ML models on 12 completely unseen materials which are recently reported in the literature.[50–59] The predictions which are within 50% of the literature-reported value are indicated using solid circles and those which are outside of 50% are denoted using open circles.

this DFT-trained ML forcefield is generic and it could be employed to predict other material properties of a given system. However, this trained forcefield is not transferable and specific to a given material(s) (for which it is trained). Further, one needs to look into the computational cost of data generation for this forcefield training compared to the speedup obtained from employing this forcefield for studying thermal properties.

Alternatively, ML can be used to explore the chemical/configuration space of a given material system to identify hidden trends and later employ the knowledge gained in this exploration to identify configurations with desired properties. The objective in this approach is to learn the property trends over a large configuration space (potentially exponentially large), which is otherwise difficult to obtain. Examples of such applications include the study by Hu *et al.*[27] to minimize coherent conduction across aperiodic interfaces and the study by Visaria and Jain[30] to explore high/low thermal κ materials from exponentially large search of graphene composed of light/heavy carbon atoms.

Finally, the ML can also be used to directly predict κ from the material structure or some other intermediate thermal/material property (such as heat capacity, speed of sound, etc.). While the direct prediction of κ from the structure is a holy grail for the discovery of low/high κ solids for various applications, it is currently limited due to the availability of only a sparse amount of data, which is insufficient in exploring the entire chemical space of inorganic materials. While approaches such as transfer learning are useful in such scenarios, our study suggests that the application of such approaches for κ have, so far, resulted in the best MAE($\log(\kappa)$) of 0.23 and $R^2(\log(\kappa))$ of 0.81 with MAPE of around 55%.

V. CONCLUSIONS

In summary, we explored the possibility of end-to-end structure to thermal conductivity prediction of materials using various machine learning approaches. Due to the scarcity of available thermal conductivity data, we first carried out high-throughput first-principles-based Boltzmann transport equation-driven prediction of the thermal conductivity to augment the currently available dataset from a current size of 166 to 398 materials. We tested various performance evaluation metrices and found that all linear-scale evaluation metrices are biased either toward under-prediction or toward high-value datapoints. We find that literature-reported best machine learning models are prone to over-fitting due to scarcity of data and we proposed a new graph-based network architecture capable of delivering more regularized and consistent performance across considered smaller size thermal conductivity datasets. Transfer learning by first training the models on larger size, lower accuracy dataset improves model performance but the best-obtained performance from considered models is limited to mean absolute percentage error of $\sim 60\%$, even on an augmented dataset (which has more than twice as many entries as the largest reported datasets available in the literature) with the newly proposed graph model (which delivers superior performance compared to all other models); thus suggesting that while machine learning can be used for initial screening of materials, the currently obtainable accuracy for end-to-end thermal conductivity prediction is limited to 50%–60%. The further improvement in performance beyond this requires larger thermal conductivity datasets and/or incorporation of intermediate atomic/thermal properties (such as thermal displacements of atoms) in the machine learning model.

ACKNOWLEDGMENTS

The authors acknowledge financial support from the National Supercomputing Mission, Government of India (Grant No. DST/NSM/R&D-HPC-Applications/2021/10), Core Research Grant, Science & Engineering Research Board, India (Grant No. CRG/2021/000010), and Nano Mission, Government of India, DST/NM/NS/2020/340. The calculations are carried out on the SpaceTime-II supercomputing facility of IIT Bombay and the PARAM Sanganak supercomputing facility of IIT Kanpur. The authors acknowledge useful discussions with Shravan Godse at IIT Bombay.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Yagyank Srivastava: Conceptualization (equal); Data curation (equal); Formal analysis (lead); Methodology (lead); Software (lead);

Validation (lead); Visualization (lead); Writing – original draft (equal); Writing – review & editing (equal). **Ankit Jain:** Conceptualization (equal); Data curation (equal); Funding acquisition (equal); Investigation (equal); Methodology (equal); Project administration (equal); Resources (equal); Supervision (equal); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1]G. A. Slack, in *CRC Handbook of Thermoelectrics*, edited by D. M. Rowe (CRC, Boca Raton, 1995) pp. 407–440.
[2]D. R. Clarke, Surf. Coat. Technol. **163**, 67 (2003).
[3]A. J. Minnich, M. S. Dresselhaus, F. Ren, and G. Chen, Energy Environ. Sci. **2**, 466 (2009).
[4]L. Lindsay and C. A. Polanco, *Handbook of Materials Modeling* (Springer Intl. Publishing, 2018), pp. 1–31.
[5]C. Dames and G. Chen, *Thermoelectrics Handbook: Macro to Nano* (CRC Press, 2005), pp. 421–4211, Chap. 42.
[6]L. Lindsay, C. Hua, X. Ruan, and S. Lee, Mater. Today Phys. **7**, 106 (2018).
[7]J. M. Ziman, *Electrons and Phonons* (Oxford University Press, Clarendon, Oxford, 1960).
[8]K. Esfarjani and H. T. Stokes, Phys. Rev. B **77**, 144112 (2008).
[9]K. Esfarjani, G. Chen, and H. T. Stokes, Phys. Rev. B **84**, 085204 (2011).
[10]A. J. McGaughey, A. Jain, H. Y. Kim, and B. Fu, J. Appl. Phys. **125**, 11101 (2019).
[11]L. Lindsay, D. A. Broido, and T. L. Reinecke, Phys. Rev. B **87**, 165201 (2013).
[12]A. Jain, Phys. Rev. B **102**, 201201 (2020).
[13]Y. Xia, V. I. Hegde, K. Pal, X. Hua, D. Gaines, S. Patel, J. He, M. Aykol, and C. Wolverton, Phys. Rev. X **10**, 041029 (2020).
[14]H. Miyazaki, T. Tamura, M. Mikami, K. Watanabe, N. Ide, O. M. Ozkendir, and Y. Nishino, Sci. Rep. **11**, 1 (2021).
[15]J. He, Y. Xia, W. Lin, K. Pal, Y. Zhu, M. G. Kanatzidis, and C. Wolverton, Adv. Funct. Mater. **32**, 2108532 (2021).
[16]K. Pal, Y. Xia, J. Shen, J. He, Y. Luo, M. G. Kanatzidis, and C. Wolverton, npj. Comput. Mater. **7**, 1 (2021).
[17]A. D. Sendek, E. D. Cubuk, E. R. Antoniuk, G. Cheon, Y. Cui, and E. J. Reed, arXiv:1808.02470 (2018).
[18]H. Wei, S. Zhao, Q. Rong, and H. Bao, Int. J. Heat Mass Transfer **127**, 908 (2018).
[19]T. Wang, C. Zhang, H. Snoussi, and G. Zhang, Adv. Funct. Mater. **30**, 1906041 (2020).
[20]A. L. Moore and L. Shi, Mater. Today **17**, 163 (2014).
[21]E. Pop, Nano. Res. **3**, 147 (2010).
[22]L. Lu, M. Dao, P. Kumar, U. Ramamurty, G. E. Karniadakis, and S. Suresh, Proc. Natl. Acad. Sci. U.S.A. **117**, 7052 (2020).
[23]E. J. Kautz, A. R. Hagen, J. M. Johns, and D. E. Burkes, Comput. Mater. Sci. **161**, 107 (2019).
[24]C. Réda, E. Kaufmann, and A. Delahaye-Duriez, Comput. Struct. Biotech. J. **18**, 241 (2020).
[25]X. Wan, W. Feng, Y. Wang, H. Wang, X. Zhang, C. Deng, and N. Yang, Nano Lett. **19**, 3387 (2019).
[26]K. Pal, C. W. Park, Y. Xia, J. Shen, and C. Wolverton, npj. Comput. Mater. **8**, 48 (2022).
[27]R. Hu, S. Iwamoto, L. Feng, S. Ju, S. Hu, M. Ohnishi, N. Nagai, K. Hirakawa, and J. Shiomi, Phys. Rev. X **10**, 021050 (2020).
[28]A. Rodriguez, C. Lin, C. Shen, K. Yuan, M. Al-Fahdi, X. Zhang, H. Zhang, and M. Hu, Commun. Mater. **4**, 61 (2023).
[29]X. Wan, D. Ma, D. Pan, L. Yang, and N. Yang, Mater. Today Phys. **20**, 100445 (2021).
[30]D. Visaria and A. Jain, Appl. Phys. Lett. **117**, 202107 (2020).
[31]T. Zhu, R. He, S. Gong, T. Xie, P. Gorai, K. Nielsch, and J. C. Grossman, Energy Environ. Sci. **14**, 3559 (2021).
[32]Z. Liu, M. Jiang, and T. Luo, Mater. Today Phys. **28**, 100868 (2022).
[33]J. A. Reissland, *The Physics of Phonons* (John Wiley & Sons Ltd, 1973).
[34]G. P. Srivastava, *The Physics of Phonons* (Adam Hilger, Bristol, 1990).
[35]S. Baroni, S. de Gironcoli, A. Dal Corso, and P. Giannozzi, Rev. Mod. Phys. **73**, 515 (2001).
[36]P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G. L. Chiarotti, M. Cococcioni, I. Dabo, A. D. Corso, S. de Gironcoli, S. Fabris, G. Fratesi, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A. P. Seitsonen, A. Smogunov, P. Umari, and R. M. Wentzcovitch, J Phys-Condens Mat **21**, 395502 (2009).
[37]M. Schlipf and F. Gygi, Comput. Phys. Commun. **196**, 36 (2015).
[38]A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, APL Mater. **1**, 011002 (2013).
[39]D. P. Kingma and J. Ba, arXiv:1412.6980 (2014).
[40]T. Xie and J. C. Grossman, Phys. Rev. Lett. **120**, 145301 (2018).
[41]C. Chen, W. Ye, Y. Zuo, C. Zheng, and S. P. Ong, Chem. Mater. **31**, 3564 (2019).
[42]A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, *Advances in Neural Information Processing Systems 32* (Curran Associates Inc., 2019).
[43]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, and E. Duchesnay, J. Mach. Learn. Res. **12**, 2825 (2011); available at http://jmlr.org/papers/v12/pedregosa11a.htm
[44]C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*, Adaptive Computation and Machine Learning series (MIT Press, London, England, 2005).
[45]E. Schulz, M. Speekenbrink, and A. Krause, J. Math. Psychol. **85**, 1 (2018).
[46]I. Y. Miranda-Valdez, L. Viitanen, J. Mac Intyre, A. Puisto, J. Koivisto, and M. Alava, Carbohydr. Polym. **298**, 119921 (2022).
[47]B. Wehbe, M. Hildebrandt, and F. Kirchner, in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2017).
[48]L. Ward, A. Dunn, A. Faghaninia, N. E. R. Zimmermann, S. Bajaj, Q. Wang, J. H. Montoya, J. Chen, K. Bystrom, M. Dylla, K. Chard, M. Asta, K. Persson, G. J. Snyder, I. Foster, and A. Jain, Comput. Mater. Sci. **152**, 60 (2018).
[49]N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, J. Mach. Learn. Res. **15**, 1929 (2014); available at http://jmlr.org/papers/v15/srivastava14a.html
[50]A. Özden, E. Zuñiga-Puelles, J. Kortus, R. Gumeniuk, and C. Himcinschi, J. Raman Spectrosc. **54**, 84 (2023).
[51]M. Wagner-Reetz, D. Kasinathan, W. Schnelle, R. Cardoso-Gil, H. Rosner, Y. Grin, and P. Gille, Phys. Rev. B Condens. Matter Mater. Phys. **90**, 195206 (2014).
[52]K. Guo, T. Weng, Y. Jiang, Y. Zhu, H. Li, S. Yuan, J. Yang, J. Zhang, J. Luo, Y. Grin, and J.-T. Zhao, Mater. Today Phys. **21**, 100480 (2021).
[53]Q. Liu, K.-F. Liu, Q.-Q. Wang, X.-C. Liu, F. Yu, J. Liu, Y.-Y. Su, and S.-Q. Xia, Acta Mater. **230**, 117853 (2022).
[54]W. Qiu, L. Wu, X. Ke, J. Yang, and W. Zhang, Sci. Rep. **5**, 13643 (2015).
[55]E. S. Toberer, A. Zevalkink, N. Crisosto, and G. J. Snyder, Adv. Funct. Mater. **20**, 4375 (2010).
[56]W. G. Zeier, A. Zevalkink, E. Schechtel, W. Tremel, and G. J. Snyder, J. Mater. Chem. **22**, 9826 (2012).
[57]G. Ding, J. Carrete, W. Li, G. Y. Gao, and K. Yao, Appl. Phys. Lett. **108**, 233902 (2016).
[58]G. Ding, J. He, Z. Cheng, X. Wang, and S. Li, J. Mater. Chem. C Mater. Opt. Electron. Devices **6**, 13269 (2018).
[59]T. Yue, B. Xu, Y. Zhao, S. Meng, and Z. Dai, Phys. Chem. Chem. Phys. **24**, 4666 (2022).
[60]T. Han, J. Li, L. Liu, F. Li, and L.-W. Wang, New J. Phys. **25**, 093007 (2023).