Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2023, **2**, 651

Synthetic data enable experiments in atomistic machine learning

John L. A. Gardner, Zoé Faure Beaulieu and Volker L. Deringer *

Machine-learning models are increasingly used to predict properties of atoms in chemical systems. There have been major advances in developing descriptors and regression frameworks for this task, typically starting from (relatively) small sets of quantum-mechanical reference data. Larger datasets of this kind are becoming available, but remain expensive to generate. Here we demonstrate the use of a large dataset that we have "synthetically" labelled with per-atom energies from an existing ML potential model. The cheapness of this process, compared to the quantum-mechanical ground truth, allows us to generate millions of datapoints, in turn enabling rapid experimentation with atomistic ML models from the small- to the large-data regime. This approach allows us here to compare regression frameworks in depth, and to explore visualisation based on learned representations. We also show that learning synthetic data labels can be a useful pre-training task for subsequent fine-tuning on small datasets. In the future, we expect that our open-sourced dataset, and similar ones, will be useful in rapidly exploring deep-learning models in the limit of abundant chemical data.

Received 9th December 2022
Accepted 20th March 2023

DOI: 10.1039/d2dd00137c

rsc.li/digitaldiscovery

Introduction

Chemical research aims to understand existing, and to discover new, molecules and materials. The vast size of compositional and configurational chemical space means that physical experiments will quickly reach their limits for these tasks.[1–3] Digital "experiments", powered by large datasets and machine learning (ML) models, provide high-throughput approaches to chemical discovery, and can help to answer questions that their physical counterparts on their own can not.[4–7] However, because ML methods generally rely on large datasets rather than on empirical physical knowledge, they require new insight into the methodology itself – one example in this context is the active research into interpretability and explainability of ML models.[8,9]

Among the central tasks in ML for chemistry is the prediction of atomistic properties as a function of a given atom's chemical environment. Atomistic ML models have now been developed to predict scalar (*e.g.*, isotropic chemical shifts),[10,11] vector (*e.g.*, dipole moments),[12] and higher-order tensor properties (*e.g.*, the dielectric response).[13] ML methods are also increasingly enabling accurate, large-scale atomistic simulations based on the "learning" of a given quantum-mechanical potential-energy surface. Widely used approaches for ML interatomic potential models include neural networks (NNs),[14–17] kernel-based methods,[18,19] and linear fitting.[20,21] The most suitable choice out of these options may depend on the task and chemical domain.[22] For instance, the ability of NNs to scalably learn compressed, hierarchical, and meaningful representations has allowed them to converge to "chemical accuracy" in the small-molecule setting on the established QM9 dataset.[23–26]

When exploring a new chemical system for which there is no established, large dataset, it is not necessarily obvious which model class will be suitable for a given task, or how a model will perform. Unfortunately, creating the high-quality, quantum-mechanically accurate data needed to train such ML models is very expensive. For instance, using density-functional theory (DFT) to generate and label the 1.2 million structures within the OC20 database required the use of large-scale compute resources, and millions of CPU hours.[27] This cost often limits the size of dataset available when exploring different model classes on new chemical domains, favouring simpler models with high data economy over more complex ones that benefit from large data quantities.

Here we demonstrate the use of synthetic data labels, obtained from an existing ML potential model (Fig. 1), as a means to sidestep the high computational cost of quantum-accurate labelling that would otherwise be required for experimenting with atomistic ML approaches. Concretely, we introduce an open-sourced dataset containing 22.9 million atomic environments drawn from ML-driven molecular-dynamics (MD) simulations of diverse disordered carbon structures, subsequently labelled in less than a day on local, consumer-level compute. The size of this dataset enables us to study the behaviour of different ML models in the small- and large-data limits.

Department of Chemistry, Inorganic Chemistry Laboratory, University of Oxford, Oxford OX1 3QR, UK. E-mail: volker.deringer@chem.ox.ac.uk

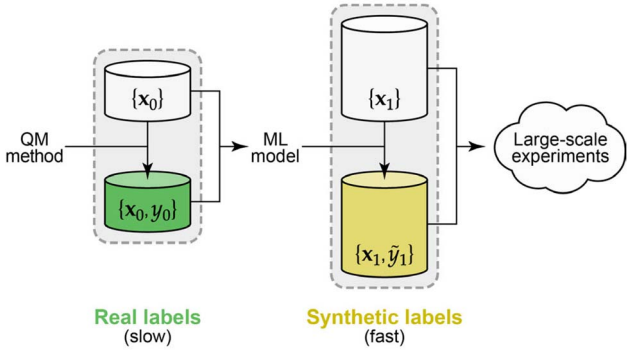

Fig. 1 Synthetic data for atomistic ML. *Left:* Quantum-mechanical (QM) data are used to label a set of structures, $\mathbf{x}_0$, with energy and force data, y_0, and these serve as input for an ML model of the potential-energy surface. *Right:* QM labels are expensive, and so we here use an existing ML model to cheaply generate and label a much larger dataset. The data in this set are "synthetic" as they are not labelled with the ground-truth QM method itself, yet represent its behaviour (note that whilst the QM method describes energies and forces on atoms, our synthetic dataset is labelled only with per-atom energies in the present study).

Dataset

Our dataset consists of 546 independent MD trajectories describing the melt-quenching and thermal annealing of elemental carbon. The development of ML potentials for carbon[28,31–34] and their application to scientific problems[35–37] have been widely documented in the literature, and the existence of established potentials such as C-GAP-17 (ref. 28) means that there is a direct route for creating synthetic data.

Initial randomised configurations of 200 atoms per cell at varying densities, from 1.0 to 3.5 g cm^{-3} in 0.1 g cm^{-3} increments, were generated using ASE.[38] Each structure then underwent an MD simulation driven by C-GAP-17, as implemented in LAMMPS.[39] First, each structure was melted at 9000 K for 5 ps before being quenched to 300 K over a further 5 ps. Second, each structure was reheated to a specific temperature at which it was annealed for 100 ps, before finally being cooled back down to 300 K over 50 ps. The annealing temperatures ranged from 2000 to 4000 K in 100 K increments. These protocols are in line with prior quenching-and-annealing type simulations with empirical and machine-learned potentials.[40–43]

The resulting database captures a wide variety of chemical environments, including graphitic structures, buckyball-esque clusters, grains of cubic and hexagonal diamond, and tetrahedral amorphous carbon. Every atom in the dataset was labelled using the C-GAP-17 potential, which predicts per-atom energies as a function of a given atom's environment.[28] Fig. 2a shows the distribution of these energies in the dataset, categorised in a simplified manner by their coordination number: "sp" as in carbon chains ($N = 2$), "sp^2" as in graphite ($N = 3$), and "sp^3" as in diamond ($N = 4$). The energies for the sp^2 and sp^3 environments are rather similar, consistent with the very similar energy of graphite and diamond; those for sp atoms are notably higher.

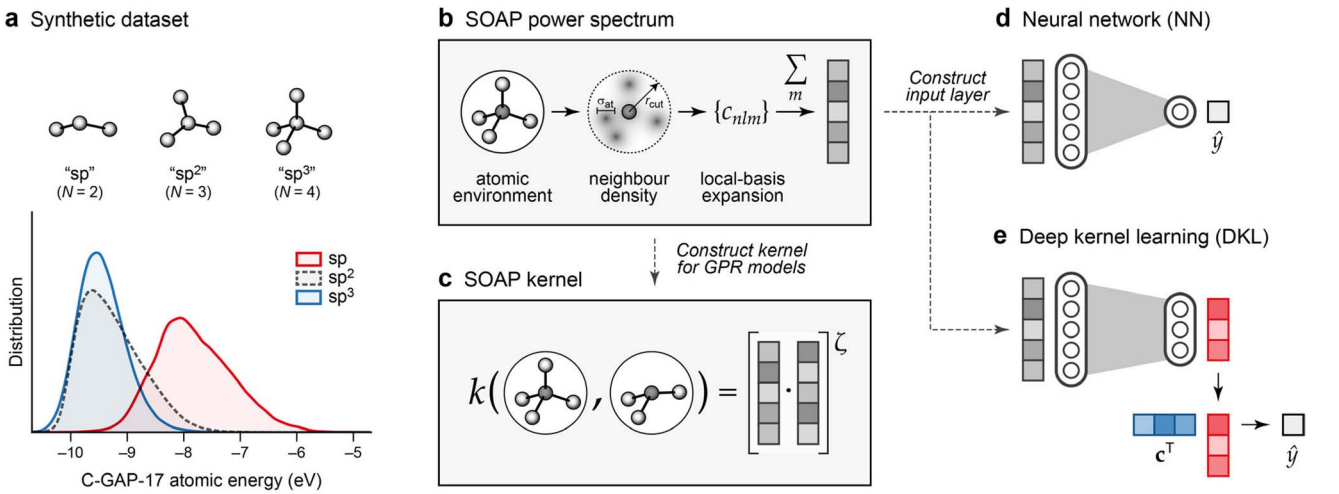

Fig. 2 Overview of ML approaches used in the present work. (a) A synthetic dataset of atomic energies, predicted by the C-GAP-17 model,[28] for different categories of carbon environments as sketched. The distributions are shown by kernel density estimates, normalised to the same value for each (note there are much fewer sp than sp^2 atoms overall). Energies are referenced to that of a free atom. (b) Smooth Overlap of Atomic Positions (SOAP) power spectrum.[29] The power-spectrum vector is an invariant fingerprint of the atomic environment, and illustrated in grey. (c) Construction of the SOAP kernel, as a dot-product of the power-spectrum vectors for two atomic environments, raised to a power of ζ. (d) Neural-network model. We use the power-spectrum vector [*cf.* panel (b)] to directly construct the input layer, and train a network to predict a new value, $\hat{y}$, from this. (e) Deep kernel learning. A neural network is used to learn a compressed representation of atomic environments, indicated in red, from the original SOAP vectors. Gaussian process regression is then used to make predictions in this compressed space, using a learned set of coefficients, $\mathbf{c}$, and the similarity of a new point to each entry in the data set. Panel (b) is adapted from ref. 30, originally published under a CC BY licence (https://creativecommons.org/licenses/by/4.0/).

Methods

Structural descriptors

We describe ("featurise") atomic environments using the Smooth Overlap of Atomic Positions (SOAP) technique.[29] SOAP is based on the idea of a local-basis expansion of the atomic neighbour density and subsequent construction of a rotationally invariant power spectrum (Fig. 2b).[29] Initially developed as a similarity measure between pairs of local neighbourhood densities, SOAP can also provide a descriptor of a single local environment, and be used as input to other ML techniques.[44–47] In the present work, we use SOAP power-spectrum vectors in two ways: to construct kernel matrices for Gaussian process regression (GPR), and as a base from which to learn richer and compressed descriptions using neural network models (Fig. 2c–e).

The SOAP descriptor is controlled by four (hyper-) parameters.[29] Two convergence parameters, n_{max} and l_{max}, control the number of radial and angular basis functions, respectively; the radial cut-off, r_{cut}, defines the locality of the environment, and a Gaussian broadening width, σ_{at}, controls the smoothness of the atomic neighbourhood densities. Here, descriptor vectors pre-calculated using $(n_{max}, l_{max}) = (12, 6)$ led to convergence for the average value in the SOAP similarity matrix for a 200-atom structure to within 0.01%, as compared to (16, 16). Values of 3.7 Å and 0.5 Å for r_{cut} and σ_{at}, respectively, were used, in line with the settings for the C-GAP-17 model.[28]

Gaussian process regression (GPR)

GPR non-parametrically fits a probabilistic model to high-dimensional data. For a detailed introduction to GPR, see ref. 48, and for its applications in chemistry, see ref. 30. At a high level, prediction at a test point, $\mathbf{x}'$, involves calculating its similarity to each data location in the training set, $\mathbf{x}_i$, using a specified kernel, k. Each of these similarities, $k(\mathbf{x}_i,\mathbf{x}')$, then modulates a coefficient, c_i, learned during training, such that the prediction is

$$\hat{y}(\mathbf{x}') = \sum_i^N c_i \cdot k(\mathbf{x}_i, \mathbf{x}') \quad \equiv \mathbf{c} \cdot k(\mathbf{X}, \mathbf{x}'). \tag{1}$$

In this work, we evaluated $k(\mathbf{x},\mathbf{x}')$ as the dot product of the respective SOAP power-spectrum vectors, raised to the power of $\zeta = 4$ as is common practice.[30]

For an exact implementation of GPR, the time complexity for predicting $\hat{y}(\mathbf{x}')$ is $O(N)$, where N is the number of training example pairs, $\{\mathbf{x}_i, y_i\}$. However, solving for $\mathbf{c}$ during training entails an $O(N^3)$ time and $O(N^2)$ storage cost. In practical terms, this limits "full GPR" to at most a few thousand data points.[48] One approach to circumventing this unfavourable scaling is referred to as sparse GPR,[30] which only considers M representative data locations when making predictions. Prediction time complexity is therefore $O(M)$, while training entails $O(M^2N + M^3)$ time and $O(NM + M^2)$ space scaling. Provided $M \ll N$, this can significantly increase the amount of data that can be used for training in practice. In the present work, we used $M = 5000$, and varied N up to 10^6.

Neural-network (NN) models

Artificial NNs can provably represent any function given sufficient parameterisation.[49,50] For an overview of the inspiration for, workings of, and theory behind NNs, we refer to ref. 51. In brief, NNs make predictions by repeatedly applying alternating linear and non-linear transforms, parameterised by weights and biases. These are learned using backpropagation to iteratively reduce a loss function.

Throughout this work, we train NNs using standard forward and backward propagation techniques using the Adam optimiser[52] and CELU activation functions,[53] all as implemented in PyTorch.[54] The performance of a deep NN depends heavily on the choice of hyperparameters for the model architecture and training, including the depth and width of the network, and the learning rate of the optimiser. We establish optimised values for these hyperparameters using an automated process: random sweep over values, and validating on a test set (see below).

Deep kernel learning (DKL)

DKL models make predictions through the sequential application of deep NN and GPR models:[55,56] the NN takes high-dimensional data as input and outputs a compressed representation in a space where the Euclidean distance between two data points, relative to the learned length scale of the GPR model, is representative of their (dis-) similarity.

During training, the parameters of both the NN and GPR model are jointly optimised by maximising the log posterior marginal likelihood. These models were implemented using the PyTorch and GPyTorch libraries.[54,57]

Learning curves

The first result of this paper is the demonstration that our synthetic data, *viz.* ML atomic energies, can indeed be machine-learned, and how the quality of this learning depends

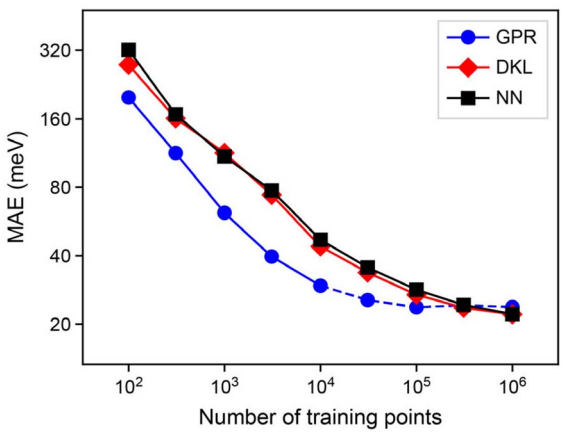

Fig. 3 Learning curves for our synthetic dataset. Mean absolute error (MAE) values for the prediction of C-GAP-17 labelled local energies, as a function of training set size ("learning curves"), for the most accurate instance of each model class. The dashed line indicates GPR models that required specialist compute (>64 GB RAM) to train.

on the size of the training dataset. In Fig. 3, we test the ability of our GPR, NN, and DKL models to learn atomic energies from the dataset of carbon structures described above. We show "learning curves" that allow us to quantify and compare the errors of the three model classes considered. In each case, mean absolute error (MAE) values are quoted as averaged using a 5-fold cross-validation procedure, where the structures from a single MD trajectory are dedicated completely to either the training or the test set, to avoid training example data leakage.[58] When using less than the full training set, we take a random sample from all atomic environments without replacement. When training network-based models, which require a validation set, we further split the shuffled training set using one tenth of the set, or 1000 points, whichever is lower. We note that this learning of ML-predicted data is related to the recently proposed "indirect learning" approach,[59] but it is distinctly different in that the latter does not regress per-atom energies, rather aiming to create teacher–student ML potential models.

In the low-data regime, the learning curves in Fig. 3 show the behaviour known for other atomistic ML models: the error decreases linearly on the double-logarithmic plot. There is a clear advantage of the GPR models (blue) over either network-based technique (NNs, black; DKL, red) in this regime, with 10^4 data points perhaps being representative of a specialised learning problem in quantum chemistry requiring expensive data labels. However, this effect is diminished upon moving to larger datasets. Typical ML potentials use on the order of 10^5 data points for training, and in this region the learning curve for the GPR models visibly saturates. We emphasise that we use sparse GPR, and so the actual number of points in the regression, M, is much lower than 10^5; this aspect will be discussed in the following section.

Comparing the NN and DKL models side-by-side, we find no notable advantage of DKL over regular NNs in this context – a slight gain in accuracy comes at a cost of approximately 100-fold slower prediction. In the remainder of this paper, we will therefore focus on a deeper analysis of GPR and NN models for atomistic ML, and report on numerical experiments with these two model classes.

Experiments

GPR insights

Having identified synthetic atomic energies as a "machine-learnable" and readily available target quantity (Fig. 3), we can use these synthetic data to gain further insight into GPR models. There are two important considerations when constructing sparse GPR models that we address here.

The first aspect is the choice of the number of representative points, M, that are used in the sparse GPR fit. In a full GPR setting, the fitting coefficients, $\mathbf{c}$, would be obtained at training time as

$$\mathbf{c} = (\mathbf{K}_{NN} + \mathbf{\Sigma})^{-1}\mathbf{y}, \tag{2}$$

where $\mathbf{K}_{NN}$ is a matrix of kernel similarity values between any two data locations, *viz.* $(K_{NN})_{i,j} = k(\mathbf{x}_i,\mathbf{x}_j)$, $\mathbf{\Sigma}$ is a regularisation term, and the vector $\mathbf{y}$ collects all labels in the training dataset.[30,48] In sparse GPR, as we use here, the analogous equation for obtaining the fitting coefficients reads:[18]

$$\mathbf{c} = [\mathbf{K}_{MM} + \mathbf{K}_{MN}\mathbf{\Sigma}^{-1}\mathbf{K}_{NM}]^{-1}\mathbf{K}_{MN}\mathbf{\Sigma}^{-1}\mathbf{y}, \tag{3}$$

where similarly defined kernel matrices, now of different sizes, are used to quantify similarities between individual atomic environments. The resulting coefficient vector, $\mathbf{c}$, now has length M (not N), and therefore the number of *representative* points, M, is what effectively controls the computational cost at runtime. For example, in a widely used GPR-based potential for elemental silicon, the reference database includes hundreds of thousands of atomic force components, whilst M is only 9000.[60]

In Fig. 4a, we show learning curves as in the previous section, but now for different values of M in otherwise similar GPR models. We find that the change from $M = 5000$ to $M = 10\,000$

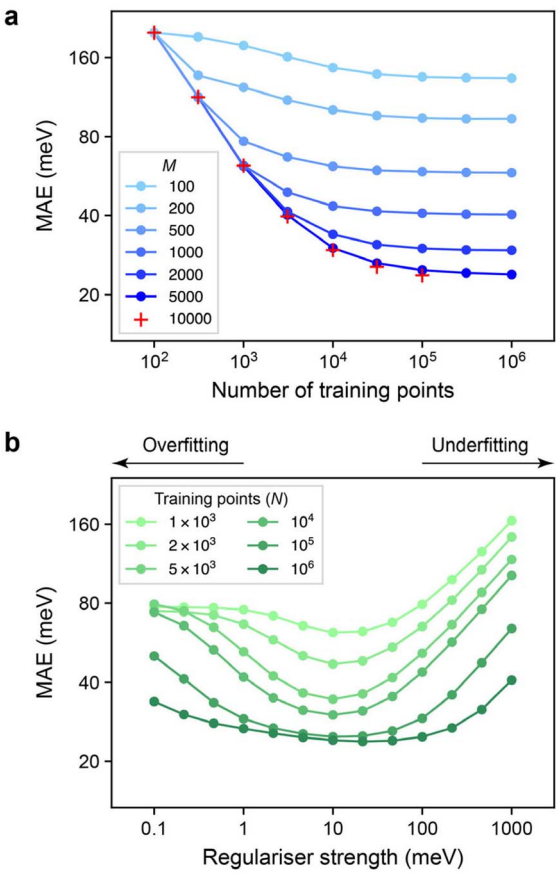

Fig. 4 Aspects of GPR models and their effect on prediction quality. (a) MAE for a series of GPR models with varied numbers of representative points, M, and training points, N. We used $M = 5000$ in the present work, limited by memory availability for $N = 10^6$. We include results for $M = 10\,000$ as far as practicable, and find that those do not lead to substantial improvements in the region tested. (b) MAE for a series of GPR models with varied regularisation terms. Too low values will cause the model to overfit to data, whereas too high values (too high "expected error") will diminish the quality of the prediction. The minimum value is found around 10 meV for most values of N, and this setting was used for all other GPR results shown in this work.

does not seem to lead to a major change any more, at least up to the range investigated.

The second aspect that we explore for our GPR models is the regularisation, controlled by the matrix $\boldsymbol{\Sigma}$ in eqn (2) and (3) above. The regularisation applied during training can be interpreted as "expected error" of the data (in the context of interatomic potentials, this might be due to accuracy and convergence limits of the quantum-mechanical training data[30]). Another interpretation is as the driving force applied during training that affects the extent to with the final fit passes through all the data points. For simplicity, we use a constant regularisation value for all atoms in the database; that is, $\boldsymbol{\Sigma} = \sigma\mathbf{I}$, with $\sigma =$ 10 meV unless noted otherwise. We note in passing that more adaptive approaches are possible, such as an individual regularisation for each atom, as exemplified before in GAP fitting.[61]

We tested the effect of varying the regularisation value, σ, over a wide range of values – the ease with which synthetic data labels are accessible means that we can rapidly fit many candidate models, both in terms of σ, and of the number of training points, N. The results are shown in Fig. 4b.

Interestingly, the dependence on the regularisation becomes less pronounced with larger N: the curves visibly flatten as one moves to larger datasets. At the same time, there is still a difference between the $N = 10^5$ and $N = 10^6$ curves in Fig. 4b, even though the learning curve itself had already "levelled off" (Fig. 3). We observe a flatter curve for the larger dataset, and a shift of the location of the minimum to higher σ, although it remains to be explored how significant the latter is. GPR is a fundamentally Bayesian technique, and so the behaviour seen in Fig. 4b can be rationalised by noting that using more data reduces the importance of any priors, in this case, of the exact value of σ. As larger and larger datasets become used for GPR-based models, tuning of the regularisation is therefore expected to become less important.

In retrospect, both plots in Fig. 4 seem to confirm settings that have been intuitively used in ML potential fitting using the GAP framework.[30] We are curious whether large-scale experiments on synthetic (proxy) data can, in the future, inform the choice of regularisation and other hyperparameters in new and more complex "real-world" GPR models for chemistry. We argue that the speed at which these atomistic GPR energy models can be fit makes this proposition attractive (for reference, training a model on 10^6 atomic environments using 5000 sparse points took 61 minutes on a mid-range dual-CPU node).

Pre-training

We now move to the discussion of neural-network methodology for atomistic ML. We hypothesised that our synthetic dataset can be used to pre-train an atomistic NN model, which can then be fine-tuned for predicting a related property. For this

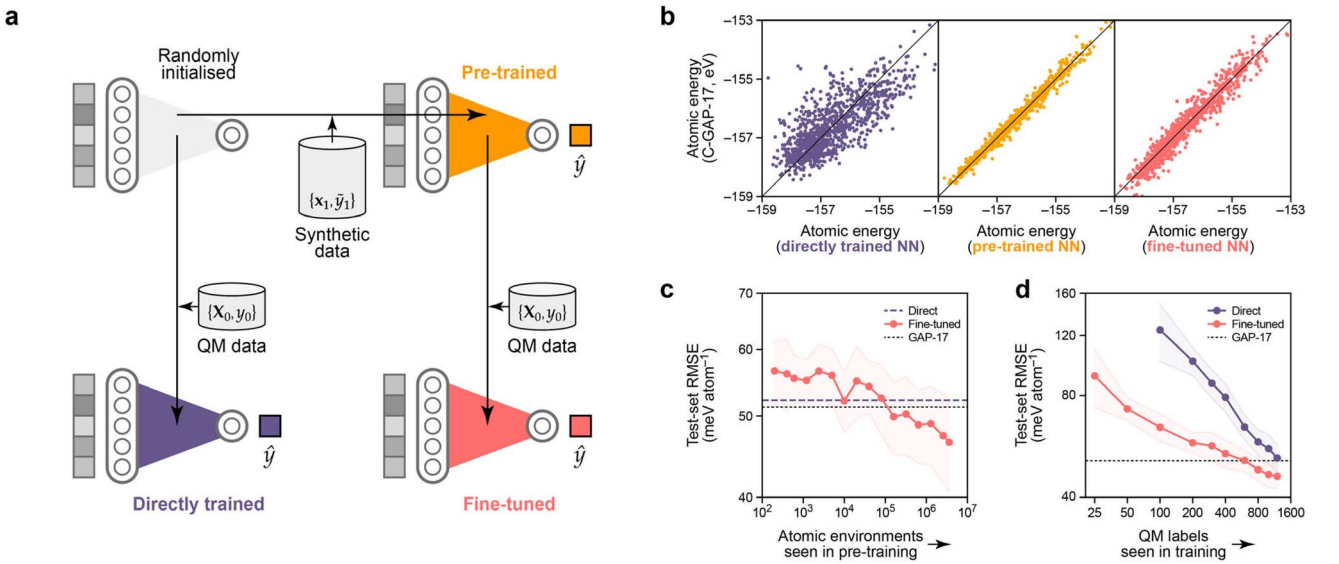

Fig. 5 Pre-training neural networks with synthetic atomistic data. (a) Schematic of the experiment. We pre-train an NN (orange) on local energies from a large synthetic dataset, **D1** ($\equiv\{\mathbf{x}_1,\tilde{y}_1\}$), then use the optimised parameters as a starting point for training ("fine-tuning") another NN (red) on quantum-mechanical ("QM") total energies from a smaller dataset, **D0**. We compare against the more conventional approach of directly training on **D0** (purple). (b) Parity plots for local (per-atom) energy predictions, testing against **D1**, *i.e.*, against the performance of C-GAP-17. From left to right: the directly trained NN (which learns to assign local energies in a different manner from GAP), the pre-trained NN (with tight correlation), and finally the fine-tuned model. 1000 data points are drawn at random from the corresponding cross-validation test set. RMSE values are given in Table 1. (c) Effect of varying the number of pre-training environments on final test-set accuracy when fine-tuned on the complete **D0**. Low numbers of pretraining environments have little effect on the final accuracy (∼ no change as compared to direct training). Increasing the number of pre-training environments beyond about 10^5 (*i.e.*, roughly the number of environments in **D0**) leads to increasing performance of the fine-tuned over the directly trained model. (d) Learning curves that show the dependence on the number of QM labels seen during training. To obtain a model with the same predictive power, in this case, ∼8× fewer QM labels are required when starting from a pre-trained NN, as compared to random initialisation. Note that we truncate the plot at 100 QM labels for direct training, but extend it to as low as 25 for the fine-tuned NNs.

approach to be useful, it needs to lead to a better final model than training an NN directly without prior information. The idea behind this experiment is sketched in Fig. 5a.

The task on which we have focused so far is to minimise the atom-wise (squared) error of our model predictions as compared to synthetic labels:

$$\underset{\lambda}{\operatorname{argmin}}\ \mathcal{L}\left(\sum_i |\tilde{\varepsilon}_i - \hat{\varepsilon}_\lambda(\mathbf{x}_i)|^2\right), \tag{4}$$

where $\tilde{\varepsilon}$ are the ML atomic energies, as labelled by C-GAP-17 and used here as synthetic training data, and $\hat{\varepsilon}$ are our model's predictions of this property. The loss function, $\mathcal{L}$, is optimised with respect to the set of model parameters, λ.

The task that we ultimately want to perform is the prediction of quantum-mechanical, per-cell energies, E: we have a dataset, **D0**, consisting of pairs $\{\mathbf{X}, E_{\mathrm{DFT}}\}$, where $\mathbf{X}$ is the set of descriptor vectors, $\mathbf{x}_i$, together describing all atoms in a given unit cell, and E_{DFT} is the per-cell energy as calculated using DFT (in this proof-of-concept, **D0** is the subset of all 64-atom amorphous carbon structures taken from the C-GAP-17 database[28]). In many currently used NN models for chemistry, this task involves predicting total, per-cell energies as a sum of local atomic energies:[14,62–64]

$$\hat{E}_c = \sum_{i\in c} \hat{\varepsilon}_\lambda(\mathbf{x}_i). \tag{5}$$

The optimisation problem then becomes

$$\underset{\lambda}{\operatorname{argmin}}\ \mathscr{L}\left(\sum_{c\in\mathbf{D0}} \left| E_{\mathrm{DFT},c} - \sum_{\mathbf{x}_i\in\mathbf{X}_c} \hat{\varepsilon}_\lambda(\mathbf{x}_i)\right|^2\right), \tag{6}$$

where $E_{\mathrm{DFT},c}$ is the ground-truth value for cell c against which the model parameters λ are optimised.

We first describe the control experiment: training a randomly initialised NN model exclusively on per-structure energies, $E_{\mathrm{DFT},c}$. The resulting model (purple in Fig. 5a) learns atomic energies that, when summed over a cell, predict per-cell energies with an average test-set RMSE of 51.4 meV per atom. This is, to within noise, the same as the original C-GAP-17 model (on its own training data!). Interestingly, the NN model trained in this way learns to partition per-cell energies into atomic contributions in a different manner to C-GAP-17: a parity plot of these shows only a loose correlation (Fig. 5b), and the RMSE is on the order of 750 meV per atom (Table 1). This non-uniqueness of local energies from NN models seems to be in keeping with previous findings.[65,66]

Table 1 Errors for different NN models (*cf.* Fig. 5a and b), tested either on atomic energies from the C-GAP-17 labelled synthetic dataset, or on total energies from DFT

	RMSE (meV per atom)		
	Directly trained	Pre-trained	Fine-tuned
Test on $\tilde{\varepsilon}$	748.8	156.7	338.9
Test on E_{DFT}	51.4	70.6	45.1

Training a new NN solely on C-GAP-17 local energies for the synthetic dataset unsurprisingly leads to a model (orange in Fig. 5) that reproduces these quantities much more closely. Starting from this pretrained model and its set of optimised parameters, and subsequently performing the same per-cell energy optimisation procedure of eqn (6), we obtain an NN (red in Fig. 5) that performs significantly better than direct training from a random initialisation in predicting per-cell energies, with a test-set RMSE of 45.1 meV per atom (Table 1). The parity plots (Fig. 5b) show that the fine-tuned network, having been originally guided by the C-GAP-17 local energies, partitions local energies in a more similar manner to C-GAP-17 as compared to the direct training approach.

We perform preliminary tests for the role of dataset size in this pre-training procedure. Fig. 5c suggests that **D1** needs to be at least as large as **D0** (in terms of number of atomic environments) in order for the pre-training approach to improve upon the accuracy from direct training on **D0** (purple dashed line). Note that we use hyperparameters that maximise accuracy when training on the full **D1** set (here, using 4 million atomic environments) for all pre-training dataset sizes investigated; thus, while Fig. 5c might seem to suggest that small amounts of pre-training are detrimental, we assume that they actually have no effect in practice. Fig. 5d shows that, when pre-training on the full **D1** set (red), $\sim 8\times$ fewer QM labels are required to achieve the same final accuracy compared to using the direct approach (purple). Thus we have shown initial evidence that learning to predict synthetic atomic energies can be a useful and meaningful "pre-training task" for chemistry.

The trends present in Fig. 5d suggest that on the order of 5000 QM data labels would close the gap between fine-tuned and directly trained models. While this amount of data would not be difficult to obtain with standard DFT approaches, there are more accurate methods available (such as periodic random-phase approximation or quantum Monte Carlo) where this amount of data would be much more expensive to generate. We therefore propose that these would be particularly interesting use cases for synthetic pre-training, especially because this technique provides the largest performance increase for low N.

We note that our pre-training can be recast as transfer learning from a lower to a higher level of quantum-aware labelling. Transfer learning for atomistic models has been demonstrated by Smith *et al.* from DFT to coupled-cluster quality,[68] and by Shui *et al.* from empirical force fields to DFT.[47] We also note that Huang *et al.* have used atomic energies to train NN-based atomistic models.[69] In a wider perspective, the pre-training of NN models is a well-documented approach in the ML literature for various applications and domains,[70–74] and it has very recently been described in the context of interatomic potential models,[47,75,76] property prediction with synthetic pre-training data,[77] and as a means to learn general-purpose representations for atomistic structure.[76]

Embedding and visualisation

We finally illustrate the usefulness of synthetic atomistic data for the visualisation of structural and chemical space. This is an

increasingly important task in ML for chemistry, leading to what are commonly referred to as "structure maps".[78] Out of the many recipes for creating such a map, a popular one entails (i) selecting a metric to define the (dis-) similarity, or distance, between two atomic environments, (ii) calculating this metric for each pair from a (representative) set of environments to create a distance matrix, and (iii) embedding this matrix onto a low-dimensional (2D or 3D) manifold. A popular instantiation of this recipe is to use the SOAP kernel as a similarity metric, and to use a non-linear dimensionality reduction technique, such as UMAP or t-SNE, to embed the distance matrix.[78,79] Such SOAP-based maps have been reported for pristine and chemically functionalised forms of carbon[37,80,81] and can help in understanding local environments – *e.g.*, in assigning the chemical character of a given atom beyond the simplified "sp"/"sp^2"/"sp^3" labels.[80]

Our present work explores the ability of an NN model to generate analogous 2D maps of chemical structure. Outputs from hidden layers of an atomistic NN can be used to visualise structural space.[82] We investigate such visualisation for a model that has been trained on the synthetic dataset introduced above, and draw comparison with earlier work on carbon.[80] We adapt the above recipe by using the Euclidean distance between NN penultimate hidden-layer representations as a dissimilarity metric, and embed the resulting matrix using UMAP,[67] a general approach not limited to chemistry.[83]

Fig. 6 shows the resulting structure maps. We use 30 000 atomic environments selected at random from the dataset presented above. The upper row (Fig. 6a–c) shows a SOAP-based UMAP embedding, colour-coded by relevant properties, whereas the lower row (Fig. 6d–f) shows a UMAP embedding derived from hidden-layer representations of an NN model. The former confirms observations made before on smaller datasets:[80] distinct types of environments, both energetically and structurally, can be discerned in the map by different colour coding. For example, there is a small island of structures with high local energy (yellow, Fig. 6a), and those correspond to the twofold-

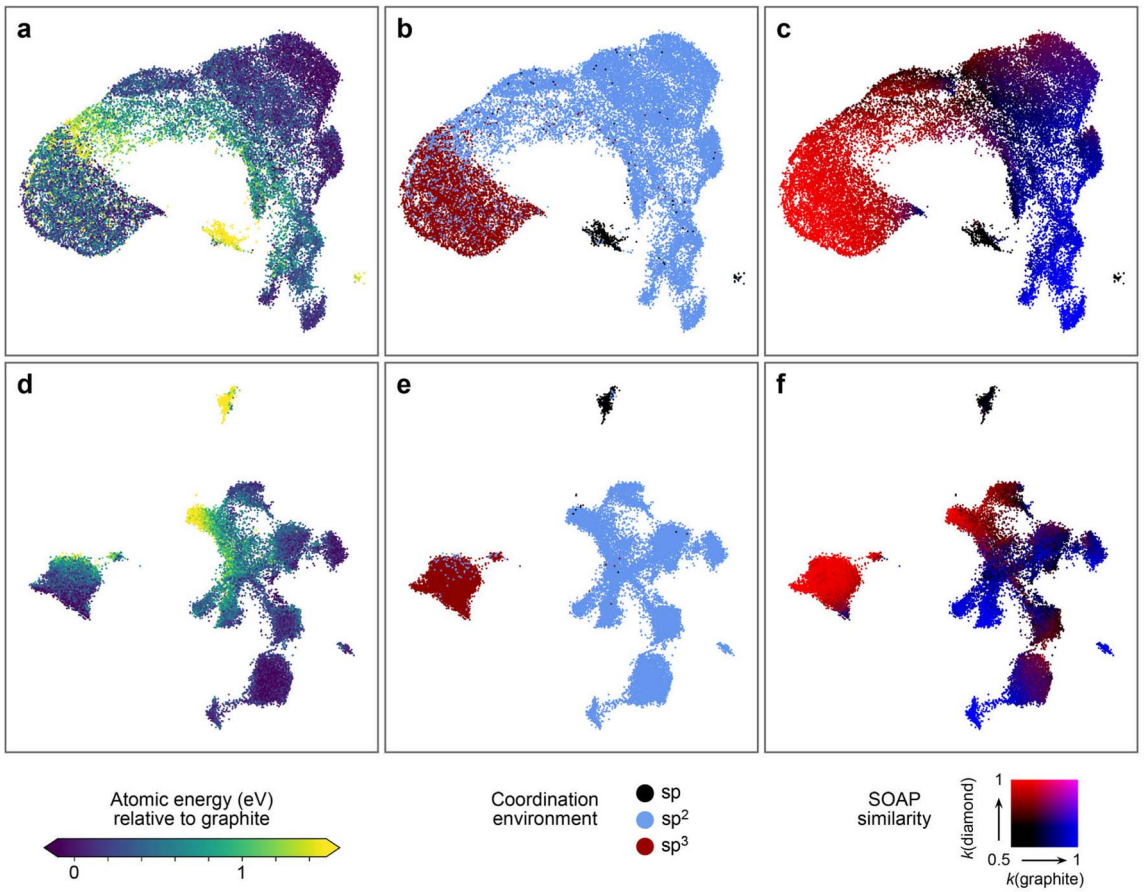

Fig. 6 UMAP projections[67] to visualise the configurational space of the synthetic dataset, as described by the original SOAP vectors (a–c), and by the compressed representations learned by an NN trained on synthetic C-GAP-17 atomic energies (d–f). From left to right, we colour-code by the C-GAP-17 atomic energy relative to graphite, by coordination environment category as determined by a 1.85 Å cut-off, and by SOAP similarity to diamond (red) and graphite (blue). Some clustering exists in the original SOAP space, although many strictly sp^2 atoms are found within the space predominantly populated by sp^3 carbon. At a local scale, the gradient in atomic energy is very noisy in this space. Compare this to the representation learned by the NN: clear clustering occurs that aligns very tightly with carbon coordination environment. Within the sp^2 region, further sub-clustering exists, each of which has clear meaning as highlighted by the SOAP similarity colour coding. The local gradient in atomic energy is much smoother, as is to be expected given the network has been trained on this quantity.

bonded "sp" environments, as seen from Fig. 6b. This observation is consistent with the energy distributions shown in Fig. 2a.

Whilst the SOAP map does therefore capture relevant aspects, the clustering in the map produced from the learned NN representations (Fig. 6d–f) is significantly more intricate, while also aligning more strongly with our understanding (or the chemistry textbook picture) of carbon atom hybridisation. Specifically, compared to the embeddings produced from SOAP descriptors, the NN space shows a much clearer separation of sp^2 *vs.* sp^3 atoms (dark red *vs.* light blue in the central panels), and it also shows sub-clustering within the sp^2 region. We can interpret this sub-clustering by colour-coding each datapoint according to the corresponding environment's SOAP similarity to both graphite (blue) and diamond (red). These results are shown on the right-hand side of Fig. 6: some of the formally sp^2 carbons are very similar to diamond-like environments, suggesting that these are in fact dangling-bond sp^3 environments, further corroborated by their high local energies. This kind of structure is not made as obvious in the SOAP map in Fig. 6c.

In Fig. 7, we show further evidence that the NN has learned a different description of local structure from the original SOAP descriptors. We performed cluster analysis in the NN-based structure map, using the BIRCH algorithm[84] to separate the data into 7 distinct clusters (colour-coded arbitrarily in Fig. 7a), and then projected the resulting cluster labels into the space of the original SOAP map (Fig. 7b). Doing so shows that atoms contained within the same cluster in NN space are not necessarily co-located in SOAP space: while the sp atoms remain isolated in the SOAP map (*cf.* Fig. 6b), the remaining clusters are clearly heavily intermixed, with some (*e.g.*, bright green) spanning most of the SOAP space. Hence, the mapping between SOAP vector and NN representation is complex and highly nonlinear, such that the NN is truly learning a new representation.

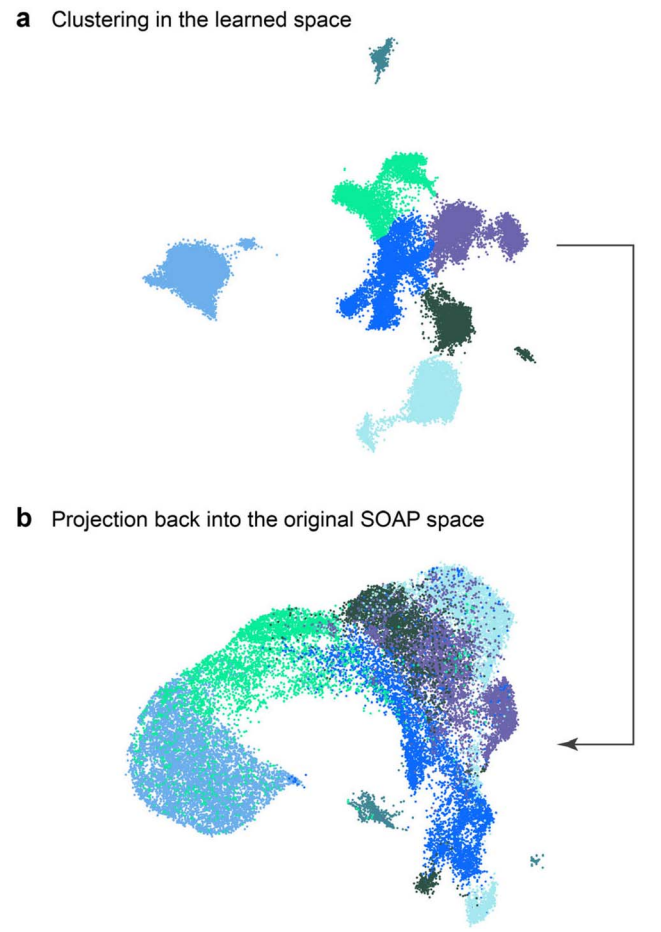

Fig. 7 UMAP embeddings showing (a) clusters found in the NN-based map of Fig. 6d–f, and (b) projection of these cluster labels into the original SOAP space. Clear mixing occurs in the latter, showing that the NN is learning a different representation than the SOAP features with which it was trained, rather than some linear recombination of these.

We therefore argue that maps based on hidden layers of atomistic NN models, trained on synthetic datasets as exemplified here, can capture aspects of both the structure and the energetics of a given material. This is not merely a consequence of the higher flexibility of NNs – in fact, an analogue to the SOAP-based maps shown herein would be the visualisation of the latent space of an autoencoder model. Instead, we here take the hidden layer following *supervised* learning, thereby automatically incorporating information about the data labels in the structure map (although the question how exactly this information is learned is deliberately left to the network to optimise). There is some similarity of this approach to principal covariates regression[85] which has been combined with kernel metrics for use in atomistic ML.[86] Here, however, the data labels can enter into the model in nonlinear form, and also the embedding of the structural information itself is more intricate. Where applicable, our findings are in line with a very recent study by Shui *et al.*, who demonstrated that pre-training can lead to more meaningful embeddings for SchNet models compared to random initialisation.[47] We suggest that maps of similar type could be explored for different systems and application problems in chemistry.

Discussion

Our study demonstrates that "synthetic" atomic energies, predicted at large scale by a machine-learning model, are themselves learnable and can be used to study the behaviour of different atomistic ML techniques. In the present work, we have compared the ability of GPR, NN, and DKL models to predict properties of chemical environments in the large-data limit, using atomic energies as a proxy for other quantities.

By means of numerical experiments, we showed that network-based models are able to learn useful representations from the original SOAP descriptors, and that these can lead to improved accuracies compared to SOAP-GPR models if the number of training data points is large. Whereas DKL can substantially outperform stand-alone NNs in some applications,[87] and has begun to be successfully applied to research questions in chemistry,[88,89] we have found that in the setting of the present work (*i.e.*, regression of large amounts of per-atom energy data), DKL models were only slightly more accurate than NNs whilst being significantly more expensive when making predictions. In comparison, SOAP-based GPR models, while slower and less accurate in the large-data regime, show

better generalisation (and thus accuracies) for small amounts of data. This finding is consistent with the marked success of sparse-GPR-based atomistic ML models on datasets of (relatively) modest size.[30]

In the present work, we have focused on learning ML atomic energies. These values do not directly correspond to any quantum-mechanical observable, and yet empirically they do appear to correlate well with local topological disorder and distortions,[90] and they can be used to drive structural exploration.[91] Irrespective of their physical interpretation (or absence thereof), we propose that ML atomic energies are a useful regression target for NN models: the compressed representations of structure learned through this task are imbued with deep, and to some extent interpretable, meaning (Fig. 6). The fact that synthetic labels are quick to generate, and the networks quick to train, suggests that this is a useful auxiliary or pre-training task to create models with "knowledge" about a chemical space. These network models could then be used to fine-tune on much smaller datasets, using their existing and general chemical knowledge to overcome the relative weakness of NN models in the low-data regime. We have shown initial evidence for this in Fig. 5, and further work is ongoing.

Appendix: Technical details

When training all NN models, we employed as means of regularisation: early stopping (by measuring performance on a validation set), dropout, and L2 weight decay. To find optimal parameters for the number of hidden layers, layer width, learning rate, batch size, dropout fraction, and weight decay magnitude, we performed a random (Hammersley) search over a broadly defined hyper-parameter space. We found that the optimal learning rate in all instances was close to the commonly used 3×10^{-4} for the Adam optimiser. 3 hidden layers, each with ≈ 800 nodes, gave the most accurate models in the limit of large data, while smaller models performed better for smaller training sets, presumably because the model size is acting as further regularisation to avoid extreme overfitting. In all instances, we found high dropout ($p \approx 0.5$) to be a more effective regulariser than weight decay.

For the NN models trained on DFT per-cell energies, we performed 10-fold cross validation, reporting error metrics as averaged over the 10 folds, and ensuring that all atomic environments for a given structure are placed in either the training, test, or validation set to avoid data leakage. When using less than the full dataset to train, we sampled N random structures from the training set without replacement. The best model trained directly on the full training set obtained errors on the test and training set amounting to 51.4 and 18.1 meV per atom, respectively, and for the fine-tuned models we obtained 45.1/22.0 meV per atom on test/train data. We note that, despite aggressive regularisation during training, this generalisation gap is large – we attribute this to the small dataset size (1200 labels) used in the fine-tuning experiment. In comparison, the corresponding generalisation gap for the $N = 10^6$ NN model in Fig. 3 is much smaller, *viz.* 23.7/22.0 meV per atom for test/train data, respectively.

Data availability

The dataset supporting the present work is provided at **https://github.com/jla-gardner/carbon-data** and a copy has been archived in Zenodo at **https://doi.org/10.5281/zenodo.7704087**. Each trajectory is supplied as a standalone "extended" XYZ file, with local energies provided as a per-atom quantity. These files can be read and processed, for example, by the Atomic Simulation Environment (ASE).[38]

Code availability

Python code, data, and Jupyter notebooks for reproducing the fine-tuning experiments (Fig. 5) are openly available at **https://github.com/jla-gardner/synthetic-fine-tuning-experiments**. Code and data for the other experiments are at **https://github.com/jla-gardner/synthetic-data-experiments**. Copies have been archived in Zenodo and are available at **https://doi.org/10.5281/zenodo.7688032** and **https://doi.org/10.5281/zenodo.7688015**, respectively.

Author contributions

J. L. A. G. and V. L. D. designed the research. J. L. A. G. fitted and evaluated all models and carried out numerical experiments. Z. F. B. carried out a key proof-of-concept study. All authors contributed to discussions. V. L. D. supervised the work. J. L. A. G. and V. L. D. wrote the paper, and all authors reviewed and approved the final version.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

We thank J. N. Foerster, A. L. Goodwin, and T. C. Nicholas for useful discussions. J. L. A. G. acknowledges a UKRI Linacre – The EPA Cephalosporin Scholarship, support from an EPSRC DTP award (EP/T517811/1), and from the Department of Chemistry, University of Oxford. V. L. D. acknowledges a UK Research and Innovation Frontier Research grant [grant number EP/X016188/1] and support from the John Fell OUP Research Fund. The authors acknowledge the use of the University of Oxford Advanced Research Computing (ARC) facility in carrying out this work (**http://doi.org/10.5281/zenodo.22558**).

References

1 J.-L. Reymond and M. Awale, Exploring Chemical Space for Drug Discovery Using the Chemical Universe Database, *ACS Chem. Neurosci.*, 2012, **3**, 649–657.
2 P. G. Polishchuk, T. I. Madzhidov and A. Varnek, Estimation of the size of drug-like chemical space based on GDB-17 data, *J. Comput.-Aided Mol. Des.*, 2013, **27**, 675–679.

3 G. Restrepo, Chemical space: Limits, evolution and modelling of an object bigger than our universal library, *Digital Discovery*, 2022, **1**, 568–585.
4 S. Curtarolo, G. L. W. Hart, M. B. Nardelli, N. Mingo, S. Sanvito and O. Levy, The high-throughput highway to computational materials design, *Nat. Mater.*, 2013, **12**, 191–201.
5 C. W. Coley, N. S. Eyke and K. F. Jensen, Autonomous Discovery in the Chemical Sciences Part I: Progress, *Angew. Chem., Int. Ed.*, 2020, **59**, 22858–22893.
6 C. W. Coley, N. S. Eyke and K. F. Jensen, Autonomous Discovery in the Chemical Sciences Part II: Outlook, *Angew. Chem., Int. Ed.*, 2020, **59**, 23414–23436.
7 S. K. Kauwe, J. Graser, R. Murdock and T. D. Sparks, Can machine learning find extraordinary materials?, *Comput. Mater. Sci.*, 2020, **174**, 109498.
8 R. Dybowski, Interpretable machine learning as a tool for scientific discovery in chemistry, *New J. Chem.*, 2020, **44**, 20914–20920.
9 F. Oviedo, J. L. Ferres, T. Buonassisi and K. T. Butler, Interpretable and Explainable Machine Learning for Materials Science and Chemistry, *Acc. Mater. Res.*, 2022, **3**, 597–607.
10 F. M. Paruzzo, A. Hofstetter, F. Musil, S. De, M. Ceriotti and L. Emsley, Chemical shifts in molecular solids by machine learning, *Nat. Commun.*, 2018, **9**, 4501.
11 Z. Chaker, M. Salanne, J.-M. Delaye and T. Charpentier, NMR shifts in aluminosilicate glasses *via* machine learning, *Phys. Chem. Chem. Phys.*, 2019, **21**, 21709–21725.
12 M. Veit, D. M. Wilkins, Y. Yang, R. A. DiStasio and M. Ceriotti, Predicting molecular dipole moments by combining atomic partial charges and atomic dipoles, *J. Chem. Phys.*, 2020, **153**, 024113.
13 A. Grisafi, D. M. Wilkins, G. Csányi and M. Ceriotti, Symmetry-Adapted Machine Learning for Tensorial Properties of Atomistic Systems, *Phys. Rev. Lett.*, 2018, **120**, 036002.
14 J. Behler and M. Parrinello, Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces, *Phys. Rev. Lett.*, 2007, **98**, 146401.
15 K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko and K.-R. Müller, SchNet: A continuous-filter convolutional neural network for modeling quantum interactions, in *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17*, Red Hook, NY, USA, 2017, pp. 992–1002.
16 J. Gasteiger, J. Groß and S. Günnemann, Directional Message Passing for Molecular Graphs, *arXiv*, 2022, preprint, DOI: 10.48550/arXiv.2003.03123.
17 W. Hu, M. Shuaibi, A. Das, S. Goyal, A. Sriram, J. Leskovec, D. Parikh and C. L. Zitnick: A Graph Neural Network for Large-Scale Quantum Calculations, *arXiv*, 2021, preprint, DOI: 10.48550/arXiv.2103.01436.
18 A. P. Bartók, M. C. Payne, R. Kondor and G. Csányi, Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons, *Phys. Rev. Lett.*, 2010, **104**, 136403.
19 S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt and K.-R. Müller, Machine learning of accurate energy-conserving molecular force fields, *Sci. Adv.*, 2017, **3**, e1603015.
20 A. P. Thompson, L. P. Swiler, C. R. Trott, S. M. Foiles and G. J. Tucker, Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials, *J. Comput. Phys.*, 2015, **285**, 316–330.
21 A. V. Shapeev, Moment Tensor Potentials: A class of systematically improvable interatomic potentials, *Multiscale Model. Simul.*, 2016, **14**, 1153–1173.
22 M. Pinheiro, F. Ge, N. Ferré, P. O. Dral and M. Barbatti, Choosing the right molecular machine learning potential, *Chem. Sci.*, 2021, **12**, 14396–14413.
23 R. Ramakrishnan, P. O. Dral, M. Rupp and O. A. von Lilienfeld, Quantum chemistry structures and properties of 134 kilo molecules, *Sci. Data*, 2014, **1**, 140022.
24 N. Lubbers, J. S. Smith and K. Barros, Hierarchical modeling of molecular energies using a deep neural network, *J. Chem. Phys.*, 2018, **148**, 241715.
25 K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko and K.-R. Müller, SchNet – a deep learning architecture for molecules and materials, *J. Chem. Phys.*, 2018, **148**, 241722.
26 O. T. Unke and M. Meuwly, PhysNet: A neural network for predicting energies, forces, dipole moments, and partial charges, *J. Chem. Theory Comput.*, 2019, **15**, 3678–3693.
27 L. Chanussot, A. Das, S. Goyal, T. Lavril, M. Shuaibi, M. Riviere, K. Tran, J. Heras-Domingo, C. Ho, W. Hu, A. Palizhati, A. Sriram, B. Wood, J. Yoon, D. Parikh, C. L. Zitnick and Z. Ulissi, Open Catalyst 2020 (OC20) Dataset and Community Challenges, *ACS Catal.*, 2021, **11**, 6059–6072.
28 V. L. Deringer and G. Csányi, Machine learning based interatomic potential for amorphous carbon, *Phys. Rev. B*, 2017, **95**, 094203.
29 A. P. Bartók, R. Kondor and G. Csányi, On representing chemical environments, *Phys. Rev. B: Condens. Matter Mater. Phys.*, 2013, **87**, 184115.
30 V. L. Deringer, A. P. Bartók, N. Bernstein, D. M. Wilkins, M. Ceriotti and G. Csányi, Gaussian Process Regression for Materials and Molecules, *Chem. Rev.*, 2021, **121**, 10073–10141.
31 R. Z. Khaliullin, H. Eshet, T. D. Kühne, J. Behler and M. Parrinello, Graphite-diamond phase coexistence study employing a neural-network mapping of the *ab initio* potential energy surface, *Phys. Rev. B: Condens. Matter Mater. Phys.*, 2010, **81**, 100103.
32 P. Rowe, V. L. Deringer, P. Gasparotto, G. Csányi and A. Michaelides, An accurate and transferable machine learning potential for carbon, *J. Chem. Phys.*, 2020, **153**, 034702.
33 J. T. Willman, A. S. Williams, K. Nguyen-Cong, A. P. Thompson, M. A. Wood, A. B. Belonoshko and I. I. Oleynik, Quantum accurate SNAP carbon potential for MD shock simulations, *AIP Conf. Proc.*, 2020, **2272**, 070055.
34 Y. Shaidu, E. Küçükbenli, R. Lot, F. Pellegrini, E. Kaxiras and S. de Gironcoli, A systematic approach to generating

accurate neural network potentials: The case of carbon, *npj Comput. Mater.*, 2021, **7**, 1–13.

35 F. L. Thiemann, P. Rowe, A. Zen, E. A. Müller and A. Michaelides, Defect-Dependent Corrugation in Graphene, *Nano Lett.*, 2021, **21**, 8143–8150.

36 B. Karasulu, J.-M. Leyssale, P. Rowe, C. Weber and C. de Tomas, Accelerating the prediction of large carbon clusters *via* structure search: Evaluation of machine-learning and classical potentials, *Carbon*, 2022, **191**, 255–266.

37 D. Golze, M. Hirvensalo, P. Hernández-León, A. Aarva, J. Etula, T. Susi, P. Rinke, T. Laurila and M. A. Caro, Accurate Computational Prediction of Core-Electron Binding Energies in Carbon-Based Materials: A Machine-Learning Model Combining Density-Functional Theory and GW, *Chem. Mater.*, 2022, **34**, 6240–6254.

38 A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng and K. W. Jacobsen, The atomic simulation environment—a Python library for working with atoms, *J. Phys.: Condens. Matter*, 2017, **29**, 273002.

39 A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott and S. J. Plimpton, LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, *Comput. Phys. Commun.*, 2022, **271**, 108171.

40 R. C. Powles, N. A. Marks and D. W. M. Lau, Self-assembly of sp^2-bonded carbon nanostructures from amorphous precursors, *Phys. Rev. B: Condens. Matter Mater. Phys.*, 2009, **79**, 075430.

41 C. de Tomas, I. Suarez-Martinez, F. Vallejos-Burgos, M. J. López, K. Kaneko and N. A. Marks, Structural prediction of graphitization and porosity in carbide-derived carbons, *Carbon*, 2017, **119**, 1–9.

42 V. L. Deringer, C. Merlet, Y. Hu, T. H. Lee, J. A. Kattirtzi, O. Pecher, G. Csányi, S. R. Elliott and C. P. Grey, Towards an atomistic understanding of disordered carbon electrode materials, *Chem. Commun.*, 2018, **54**, 5988–5991.

43 Y. Wang, Z. Fan, P. Qian, T. Ala-Nissila and M. A. Caro, Structure and Pore Size Distribution in Nanoporous Carbon, *Chem. Mater.*, 2022, **34**, 617–628.

44 E. Kocer, J. K. Mason and H. Erturk, A novel approach to describe chemical environments in high-dimensional neural network potentials, *J. Chem. Phys.*, 2019, **150**, 154102.

45 M. Karamad, R. Magar, Y. Shi, S. Siahrostami, I. D. Gates and A. Barati Farimani, Orbital graph convolutional neural network for material property prediction, *Phys. Rev. Mater.*, 2020, **4**, 093801.

46 D. Xia, N. Li, P. Ren and X. Wen, Prediction Of Material Properties By Neural Network Fusing The Atomic Local Environment And Global Description: Applied To Organic Molecules And Crystals, *E3S Web Conf.*, 2021, **267**, 02059.

47 Z. Shui, D. S. Karls, M. Wen, I. A. Nikiforov, E. B. Tadmor and G. Karypis, Injecting Domain Knowledge from Empirical Interatomic Potentials to Neural Networks for Predicting Material Properties, *arXiv*, 2022, preprint, DOI: **10.48550/arXiv.2210.08047**.

48 C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning, Adaptive Computation and Machine Learning*, The MIT Press, Cambridge, MA, 2006.

49 A. Barron, Universal approximation bounds for superpositions of a sigmoidal function, *IEEE Trans. Inf. Theory*, 1993, **39**, 930–945.

50 Y. LeCun, Y. Bengio and G. Hinton, Deep learning, *Nature*, 2015, **521**, 436–444.

51 J. Schmidhuber, Deep learning in neural networks: An overview, *Neural Networks*, 2015, **61**, 85–117.

52 D. P. Kingma and J. Ba, A Method for Stochastic Optimization, *arXiv*, 2017, preprint, DOI: **10.48550/arXiv.1412.6980**.

53 J. T. Barron, Continuously Differentiable Exponential Linear Units, *arXiv*, 2017, preprint, DOI: **10.48550/arXiv.1704.07483**.

54 A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga and A. Lerer, Automatic differentiation in PyTorch, *NIPS 2017 Autodiff Workshop*, 2017.

55 A. G. Wilson, Z. Hu, R. Salakhutdinov and E. P. Xing, Deep Kernel Learning, *arXiv*, 2015, preprint, DOI: **10.48550/arXiv.1511.02222**.

56 A. G. Wilson, Z. Hu, R. Salakhutdinov and E. P. Xing, Stochastic Variational Deep Kernel Learning, *arXiv*, 2016, preprint, DOI: **10.48550/arXiv.1611.00336**.

57 J. Gardner, G. Pleiss, K. Q. Weinberger, D. Bindel and A. G. Wilson, GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration, in *Advances in Neural Information Processing Systems*, 2018, vol. 31.

58 J. D. Morrow, J. L. A. Gardner and V. L. Deringer, How to validate machine-learned interatomic potentials, *J. Chem. Phys.*, 2023, **158**, 121501.

59 J. D. Morrow and V. L. Deringer, Indirect learning and physically guided validation of interatomic potential models, *J. Chem. Phys.*, 2022, **157**, 104105.

60 A. P. Bartók, J. Kermode, N. Bernstein and G. Csányi, Machine Learning a General-Purpose Interatomic Potential for Silicon, *Phys. Rev. X*, 2018, **8**, 041048.

61 J. George, G. Hautier, A. P. Bartók, G. Csányi and V. L. Deringer, Combining phonon accuracy with high transferability in Gaussian approximation potential models, *J. Chem. Phys.*, 2020, **153**, 044104.

62 J. S. Smith, O. Isayev and A. E. Roitberg, ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost, *Chem. Sci.*, 2017, **8**, 3192–3203.

63 L. Zhang, J. Han, H. Wang, R. Car and E. Weinan, Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics, *Phys. Rev. Lett.*, 2018, **120**, 143001.

64 S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt and B. Kozinsky, E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials, *Nat. Commun.*, 2022, **13**, 2453.

65 M. Eckhoff and J. Behler, From Molecular Fragments to the Bulk: Development of a Neural Network Potential for MOF-5, *J. Chem. Theory Comput.*, 2019, **15**, 3793–3809.

66 D. Yoo, K. Lee, W. Jeong, D. Lee, S. Watanabe and S. Han, Atomic energy mapping of neural network potential, *Phys. Rev. Mater.*, 2019, **3**, 093802.

67 L. McInnes, J. Healy and J. Melville, Uniform Manifold Approximation and Projection for Dimension Reduction, *arXiv*, 2020, preprint, DOI: **10.48550/arXiv.1802.03426**.

68 J. S. Smith, B. T. Nebgen, R. Zubatyuk, N. Lubbers, C. Devereux, K. Barros, S. Tretiak, O. Isayev and A. E. Roitberg, Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning, *Nat. Commun.*, 2019, **10**, 2903.

69 Y. Huang, J. Kang, W. A. Goddard and L.-W. Wang, Density functional theory based neural network force fields from energy decompositions, *Phys. Rev. B*, 2019, **99**, 064103.

70 J. Pennington, R. Socher and C. Manning, Glove: Global Vectors for Word Representation, in *EMNLP*, 2014, vol. 14, pp. 1532–1543.

71 A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Commun. ACM*, 2017, **60**, 84–90.

72 D. Jha, K. Choudhary, F. Tavazza, W.-k. Liao, A. Choudhary, C. Campbell and A. Agrawal, Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning, *Nat. Commun.*, 2019, **10**, 5316.

73 R. Ri and Y. Tsuruoka, Pretraining with artificial language: Studying transferable knowledge in language models, in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022, vol. 1: Long Papers.

74 Y. Wu, F. Li and P. Liang, Insights into pre-training *via* simpler synthetic tasks, *arXiv*, 2022, preprint, DOI: **10.48550/arXiv.2206.10139**.

75 D. Zhang, H. Bi, F.-Z. Dai, W. Jiang, L. Zhang and H. Wang, DPA-1: Pretraining of Attention-based Deep Potential Model for Molecular Simulation, *arXiv*, 2022, preprint, DOI: **10.48550/arXiv.2208.08236**.

76 X. Gao, W. Gao, W. Xiao, Z. Wang, C. Wang and L. Xiang, Supervised Pretraining for Molecular Force Fields and Properties Prediction, *arXiv*, 2022, preprint, DOI: **10.48550/arXiv.2211.14429**.

77 I. V. Volgin, P. A. Batyr, A. V. Matseevich, A. Y. Dobrovskiy, M. V. Andreeva, V. M. Nazarychev, S. V. Larin, M. Y. Goikhman, Y. V. Vizilter, A. A. Askadskii and S. V. Lyulin, Machine Learning with Enormous "Synthetic" Data Sets: Predicting Glass Transition Temperature of Polyimides Using Graph Convolutional Neural Networks, *ACS Omega*, 2022, **7**, 43678–43691.

78 B. Cheng, R.-R. Griffiths, S. Wengert, C. Kunkel, T. Stenczel, B. Zhu, V. L. Deringer, N. Bernstein, J. T. Margraf, K. Reuter and G. Csanyi, Mapping Materials and Molecules, *Acc. Chem. Res.*, 2020, **53**, 1981–1991.

79 S. De, A. P. Bartók, G. Csányi and M. Ceriotti, Comparing molecules and solids across structural and alchemical space, *Phys. Chem. Chem. Phys.*, 2016, **18**, 13754–13769.

80 M. A. Caro, A. Aarva, V. L. Deringer, G. Csányi and T. Laurila, Reactivity of Amorphous Carbon Surfaces: Rationalizing the Role of Structural Motifs in Functionalization Using Machine Learning, *Chem. Mater.*, 2018, **30**, 7446–7455.

81 B. W. B. Shires and C. J. Pickard, Visualizing Energy Landscapes through Manifold Learning, *Phys. Rev. X*, 2021, **11**, 041026.

82 J. Westermayr, F. A. Faber, A. S. Christensen, O. A. von Lilienfeld and P. Marquetand, Neural networks and kernel ridge regression for excited states dynamics of $CH_2NH_2^+$: From single-state to multi-state representations and multi-property machine learning models, *Mach. Learn.: Sci. Technol.*, 2020, **1**, 025009.

83 S. Dorkenwald, P. H. Li, M. Januszewski, D. R. Berger, J. Maitin-Shepard, A. L. Bodor, F. Collman, C. M. Schneider-Mizell, N. M. da Costa, V. Jain, Multi-Layered Maps of Neuropil with Segmentation-Guided Contrastive Learning, *bioRxiv*, 2022, preprint, DOI: **10.1101/2022.03.29.486320**.

84 T. Zhang, R. Ramakrishnan, and M. Livny, BIRCH: An efficient data clustering method for very large databases, in *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, SIGMOD '96*, New York, NY, USA, 1996, pp. 103–114.

85 S. de Jong and H. A. L. Kiers, Principal covariates regression: Part I. Theory, *Chemometrics and Intelligent Laboratory Systems Proceedings of the 2nd Scandinavian Symposium on Chemometrics*, 1992, vol. 14, pp. 155–164.

86 B. A. Helfrecht, R. K. Cersonsky, G. Fraux and M. Ceriotti, Structure-property maps with Kernel principal covariates regression, *Mach. Learn.: Sci. Technol.*, 2020, **1**, 045021.

87 C. Yu, M. Seslija, G. Brownbridge, S. Mosbach, M. Kraft, M. Parsi, M. Davis, V. Page and A. Bhave, Deep kernel learning approach to engine emissions modeling, *Data-Centric Engineering*, 2020, **1**, e4.

88 Y. Liu, M. Ziatdinov and S. V. Kalinin, Exploring Causal Physical Mechanisms *via* Non-Gaussian Linear Models and Deep Kernel Learning: Applications for Ferroelectric Domain Structures, *ACS Nano*, 2022, **16**, 1250–1259.

89 G. Sivaraman and N. E. Jackson, Coarse-Grained Density Functional Theory Predictions *via* Deep Kernel Learning, *J. Chem. Theory Comput.*, 2022, **18**, 1129–1141.

90 N. Bernstein, B. Bhattarai, G. Csányi, D. A. Drabold, S. R. Elliott and V. L. Deringer, Quantifying Chemical Structure and Machine-Learned Atomic Energies in Amorphous and Liquid Silicon, *Angew. Chen. Int. Ed.*, 2019, **58**, 7057–7061.

91 Z. El-Machachi, M. Wilson and V. L. Deringer, Exploring the configurational space of amorphous graphene with machine-learned atomic energies, *Chem. Sci.*, 2022, **13**, 13720–13731.