Science Advances

AAAS

advances.sciencemag.org/cgi/content/full/3/12/e1701816/DC1

Supplementary Materials for

Machine learning unifies the modeling of materials and molecules

Albert P. Bartók, Sandip De, Carl Poelking, Noam Bernstein, James R. Kermode, Gábor Csányi, Michele Ceriotti

Published 13 December 2017, *Sci. Adv.* **3**, e1701816 (2017)
DOI: 10.1126/sciadv.1701816

This PDF file includes:

Machine Learning Unifies the Modelling of Materials and Molecules
Supporting Materials

Albert P. Bartók,[1] Sandip De,[2,3] Carl Poelking,[4] Noam Bernstein,[5] James Kermode,[6] Gábor Csányi,[7] and Michele Ceriotti[3]

[1)] *Scientific Computing Department, Science and Technology Facilities Council, Rutherford Appleton Laboratory, Oxfordshire OX11 0QX, United Kingdom*
[2)] *National Center for Computational Design and Discovery of Novel Materials (MARVEL)*
[3)] *Laboratory of Computational Science and Modelling, Institute of Materials, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland*
[4)] *University Chemical Laboratory, University of Cambridge*
[5)] *Center for Materials Physics and Technology, U.S. Naval Research Laboratory, Washington, DC 20375, USA*
[6)] *Warwick Centre for Predictive Modelling, School of Engineering, University of Warwick, Coventry CV4 7AL, United Kingdom*
[7)] *Engineering Laboratory, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, United Kingdom*

section 1. The atom-centered GAP is equivalent to the average molecular kernel

Consider the KRR expression for the average energy per atom of a molecule A:

$$E\left(A\right)/N_A = \sum_n w_n K\left(A, A_n\right), \tag{1}$$

where $K(A, A')$ is a kernel function that measures the similarity between the molecule A and a set of reference molecules $\{A_n\}$. The weights w_n can be optimized by requiring that, for each molecule in such reference set, the energy predicted by Eqn. (1) matches that evaluated with an explicit quantum calculation, E_n. A similar expression can be written for an atom-based energy decomposition

$$\mathcal{E}\left(\mathcal{X}\right) = \sum_i \omega_i k\left(\mathcal{X}, \mathcal{X}_i\right), \tag{2}$$

with the difference that now the kernel function measures the similarity between *atomic environments* $\mathcal{X}$, and that the KRR evaluates the contribution to the total energy originating from an individual atom. This atom-based decomposition is the conventional way to define an interatomic potential, and has previously been used to create GAPs for materials [6, 44, 45].

To see how these two expressions are related to each other, consider that in a Gaussian process regression framework the kernel between two molecules is the same as the covariance between their energies $\langle E(A)E(B)\rangle = N_A N_B K(A, B)$. Similarly, the kernel between atomic environments, is the covariance between the atomic energies, $\langle \mathcal{E}(\mathcal{X}_i)\mathcal{E}(\mathcal{X}_j)\rangle = k(\mathcal{X}_j, \mathcal{X}_i)$. Under the assumptions that the energy decomposition is fully additive, so that $E(A) = \sum_{i\in A} \mathcal{E}(\mathcal{X}_i)$, one can see that

$$\begin{aligned} K(A,B) =& \frac{\langle E(A)E(B)\rangle}{N_A N_B} = \\ =& \frac{1}{N_A N_B} \sum_{i\in A, j\in B} k\left(\mathcal{X}_i, \mathcal{X}_j\right). \end{aligned} \tag{3}$$

By substituting this expression for $K(A, B)$ into Eq. 1, it is possible to transform the expression of $E(A)$ into a sum over atom-based energies as in Eq. 2. Learning molecular energies using "structure" kernels that are equal to averages of atoms-centered kernels is thus equivalent to learning an atom-based energy decomposition using kernels between atomic environments.

section 2. A SOAP-GAP potential for silicon

The configurations comprising the training set of the SOAP-GAP model for silicon are summarized in Table **S1**. The structures were generated by DFT molecular dynamics, starting from an initial structure of the given type, and using loose convergence settings of the DFT parameters. After collecting decorrelated samples, the energies, forces and virials were recalculated with tighter convergence settings of the parameters: 250 eV plane wave cutoff and a k-point density of 0.03 Å^{-1}. The PW91[46] exchange-correlation functional was used throughout. All calculations were carried with with the CASTEP package[47]. The crack tip structures were generated using an earlier GAP model that did not include those configurations. The structures with low coordination (with sp and sp^2 hybridizations) were included because it was found that without training on them the GAP model had a tendency to predict too low energies for such structures. Although not fully automated, this is motivated by the active learning approach, and the idea that to fully capture a probability distribution (here the Boltzmann distribution corresponding to the potential) it is not enough to specify where the probability is high (low energy structures) but also where it is low (the high energy structures).

We do not optimise the hyperparameters of the Gaussian process, because previous experience shows that—consistent with the Bayesian approach—our physically motivated guesses are good enough. With the database size required for the desired accuracy, the dependence of

Structure type	# atoms	# structures	# inducing points	σ_{energy}	σ_{force}	σ_{virial}
isolated atom	1	1	1	0.001	-	-
diamond	2	104	500	0.001	0.1	0.05
	16	220				
	54	110				
	128	55				
beta-tin	2	60	500	0.001	0.1	0.05
	16	220				
	54	110				
	128	55				
hexagonal	1	110	500	0.001	0.1	0.05
	8	30				
	27	30				
	64	53				
liquid	64	69	500	0.003	0.15	0.2
	128	7				
amorphous	64	31	1000	0.01	0.2	0.4
	216	128				
diamond surface (001)	144	29	500	0.001	0.1	0.05
diamond surface (110)	108	26	500	0.001	0.1	0.05
diamond surface (111)						
unreconstructed	96	47	500			
adatom	146	11	250	0.001	0.1	0.05
Pandey	96	50	500			
DAS 3x3 unrelaxed	52	1	100			
diamond vacancy	63	100	500	0.001	0.1	0.05
	215	111				
diamond divacancy	214	78	500	0.001	0.1	0.05
diamond interstitial	217	115	500	0.001	0.1	0.05
small (110) crack tip	200	7	500	0.001	0.1	0.05
small (111) crack tip	192	10	500	0.001	0.1	0.05
screw dislocation core	144	19	200	0.001	0.1	0.05
sp^2 bonded	8	51	200	0.001	0.1	0.05
sp bonded	4	100	200	0.01	0.2	0.4
Total	169455	2148	8451			

table S1. **Summary of the database for the silicon model.** The total number of atoms corresponds to the entire database. The fitted potential has the unique label `GAP_2017_5_20_60_4_23_20_512`

the fit to the hyperparameters are quite weak. The locality of silicon is defined by the decay of the density matrix, and prior calculations indicate that force errors below 0.1 eV/Å are achievable with a cutoff of around 5 Å. The width parameter of the Gaussian functions that make up the neighbour density was 0.5 Å, close to the atomic unit of 1 Bohr, which is the typical length scale over which the potential energy varies. The truncation of the spherical harmonic expansion is a tradeoff between computational efficiency and accuracy - we typically fit a potential using tight tolerances, and as a last step, reduce the number of basis components as much as possible without compromising the accuracy. The regularisation parameters in the Gaussian process correspond to the expected accuracy, and are determined by the above locality criterion for the forces, and the estimated errors in the total energy and the virial due to the finite k-point sampling in the DFT calculations. Some high energy configurations (e.g. liquid and sp-bonded) have larger regularisation parameters.

The model is a sum of two terms. In addition to the SOAP-GAP term, we used a simple pair potential, parametrised to reproduce the dissociation and close-range repulsion behaviour of the Si dimer. The main purpose of the pair model is to augment the GPR model at short bond distances, where the energy scale is much larger compared to the attraction of interatomic bonding.

The options to the GAP fitting program to generate the SOAP-GAP model were

```
at_file=all_data.xyz gap={soap l_max=12 n_max
    =10 atom_sigma=0.5 zeta=4 cutoff=5.0
    cutoff_transition_width=1.0 central_weight
    =1.0 config_type_n_sparse={divacancy:500:
    interstitial:500:crack_110_1-10:500:
    surface_111:500:surface_110:500:sp2:200:sp
    :200:crack_111_1-10:500:dia:500:
    isolated_atom:1:bt:500:screw_disloc:200:sh
    :500:liq:500:surface_001:500:amorph:1000:
    surface_111_pandey:500:vacancy:500:111
    adatom:250:surface_111_3x3_das:100} delta
    =3.0 f0=0.0 covariance_type=dot_product
    sparse_method=cur_points} default_sigma
    ={0.001 0.1 0.05 0.0} config_type_sigma={
    liq:0.003:0.15:0.2:0.0:amorph
    :0.01:0.2:0.4:0.0:sp:0.01:0.2:0.4:0.0}
    energy_parameter_name=dft_energy
    force_parameter_name=dft_force
    virial_parameter_name=dft_virial
    config_type_parameter_name=config_type
    sparse_jitter=1.0e-8 e0_offset=2.0
    core_param_file=glue.xml core_ip_args={IP
    Glue}
```

The other interatomic potentials shown in the main paper for the DAS reconsutrctions were ReaxFF[48, 49], a Modified Embedded Atom Model[50], a Tersoff model[51] and the Stillinger-Weber model[52]. The data for the DFT curve was obtained from Solares *et al.*[53], which we shifted by a constant to match the energy of the (111) unreconstructed surface calculated using our DFT parameters and setup.

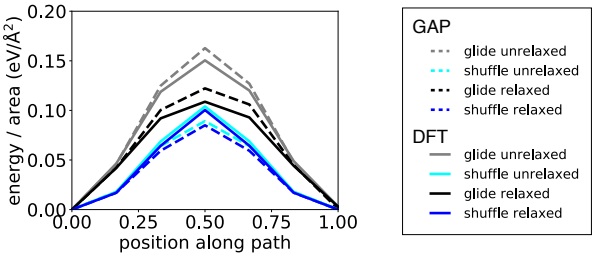

figure S1. Stacking fault energetics for the silicon model. The curves represent the energetics of paths that correspond to the formation of stacking faults in the diamond structure.

The detailed analysis of the accuracy of the GAP model in comparison to other widely used potentials will be published elsewhere. Previously published works by some of us as well as other groups indicate that nonparametric fits such as GAP are capable of reproducing with good accuracy the energetics of a wide variety of configurations that are close to those present in their training set. Our results go beyond this by (i) showing exquisite accuracy in subtle situations such as the surface reconstructions shown in the main text (including extrapolation to large system sizes), and (ii) good transferability to configurations very far from those in the training set and also away from local minima. A demonstration of the latter is in Fig. **S1**, which shows the energetics of paths the correspond to the formation of two kinds of stacking faults. The highest error is less than 15% for the glide set case, and much lower for the shuffle set - other potentials typically have 30%-50% error.

section 3. Predicting atomization energies for the GDB9 and QM7b databases

A. Computational details

DFT geometries and energies were obtained from the original GDB9 database[**?**]. To generate the PM7-optimised geometries, we started with the SMILES strings in the GDB9. We used the CORINA program (version 3.60 0066)[54] to construct three-dimensional models of the molecules and to obtain initial Cartesian coordinates. A small fraction of the molecules failed to convert, for these we used OpenBabel[55] (version 2.3.0). As a part of the conversion, hydrogen atoms were added to the structures by CORINA and OpenBabel, then the configurations were relaxed by CORINA's built-in force field and the GAFF force field[56], respectively. The resulting configurations were further relaxed at the PM7 level of semi-empirical model[57] using the MOPAC

program[58] (versions 16.043L and 17.048L).

We adopted the relaxed geometries in the GDB9 database[**?**], and we carried out geometry relaxations on the oligopeptides using the Gaussian 09 program[59]. To maintain consistence with the GDB9 database, we used the same level of theory (Density Functional Theory and the B3LYP functional[60, 61]) and the 6-31G(2df,p) basis set[62]. CCSD(T) energetics of the DFT-relaxed configurations were calculated with MOLPRO[63] (version 2012.1), using the 6-311G** basis set[66].

Unless otherwise stated, in this section we discuss learning DFT energies based on DFT-optimized geometries. While this is largely an academic exercise, given that in order to obtain DFT structures one inevitably must compute DFT energies, it has often been used as a benchmark and so it is well-suited to make our error analysis directly comparable with previous studies. Results for learning CCSD(T) energies based on DFT geometries follow very similar trends, while learning based on PM7-optimized geometries presents a different sets of challenges that are discussed in the main text.

B. Training set selection and error distribution

The GDB9 dataset contains - by construction - a relatively uneven sampling of chemical compound space, with some stoichiometries more heavily represented than others. A random selection of reference structures would give more thorough sampling of the densely populated regions, which might be advantageous to reduce the average error, but would leave extended portions of chemical space completely off the chart. An alternative approach would aim for a uniform sampling, so as to cover the margins of the distribution as well as the densely sampled regions. Farthest-point sampling (FPS) provides a simple, greedy algorithm to achieve this goal: given a set $\mathbb{S}_m = \{A_{i=1\ldots m}\}$ of molecules selected out of the overall database $\mathbb{D}$, the next molecule to be included is determined by

$$A_{m+1} = \text{argmax}_{A\in\mathbb{D}} \left[\min_{A'\in\mathbb{S}} D(A, A') \right], \qquad (4)$$

where D is the kernel-induced metric $D(A,B)^2 = K(A,A) + K(B,B) - 2K(A,B)$. Intermediate sampling methods, that balance diversity and relevance of the chosen molecules, are also possible [67]. Since Eqn. (4) relies solely on structural information, it is a practical strategy to decide where to invest computational resources to obtain a comprehensive sampling of the relevant chemical space. Fixing a maximum acceptable value of the minimum distance to the existing references, this approach also naturally extends to active learning. Whenever a new structure encountered in a simulation based on a ML potential is farther from the training set than this threshold, its energy can be computed with a high-end quantum calculation and the model be retrained on the extended reference set.

As shown in Figure **S2**, the strategy to select training points has a significant impact on the distribution of errors. Even though a FPS selection leads to a marginal increase of the MAE relative to a randomized choice, it enables a significant reduction of RMS. When studying the convergence of machine-learning methods, one should not stop at the MAE but also consider higher norms, that contain more information on the outliers, and the worst-case scenarios.

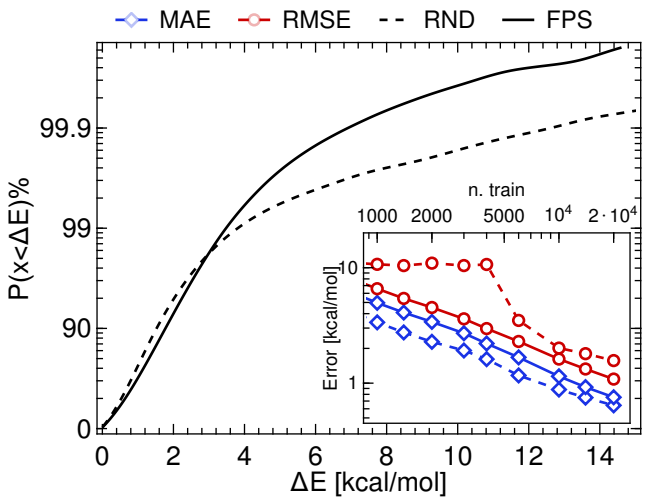

figure S2. Error distribution for the GDB9 training. Fraction of test configurations with a error smaller than a given threshold, for $n_{\text{train}} = 20,000$ training structures selected at random (dashed line) or by FPS (full line). The inset shows the learning curves resulting from the two selection strategies, comparing the mean absolute error (blue) and root mean-square error (red).

C. Training curves and hyperparameter optimization

The SOAP kernel contains several adjustable parameters – that determine its completeness, evaluation cost, and the scale of interactions [39]. The parameters to the `glosim.py` package used to generate the kernel matrix for the production calculations were

```
~/source/glosim/glosim.py datafile.xyz -n 9 -l
    9 -g 0.3 -c 3 --zeta 2 --periodic --nonorm
    --kernel average
```

The code is available from `http://cosmo-epfl.github.io`. Internally, `glosim.py` called the SOAP routines in quippy, using the template

```
"soap central_reference_all_species=F
   central_weight=1.0 covariance_sigma0=0.0
   atom_sigma="+str(g)+" cutoff="+str(c)+"
   cutoff_transition_width=0.5 n_max="+str(n)
   +" l_max="+str(l)
```

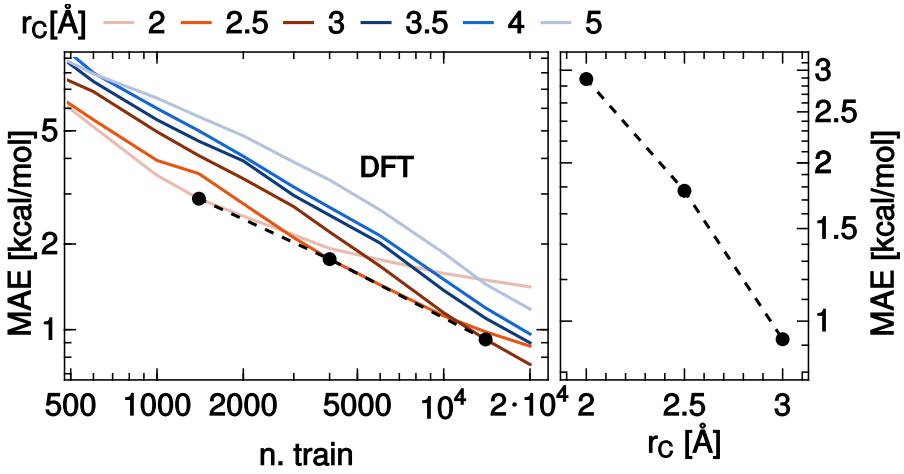

figure S3. Optimal range of interactions for learning GDB9 DFT energies. (left) Learning curves for the GDB9 dataset. 20,000 structures were selected by FPS and used for training DFT energetics, using the DFT geometries as inputs. Tests were performed on the remaining 114,000 structures. Different curves correspond to varying cutoffs in the SOAP environment selection, resulting in a different trend in the curve. Shorter cutoffs typically give smaller errors with small n_{train}, but the error saturates for larger train set size. The dashed curve highlights the envelope of the various training curves, signifying which value of r_C gives the best performance for each training set size. The same points, plotted as a function of r_C (right) give a sense of the energy scale of the interactions that can be modelled with local description over the specified range.

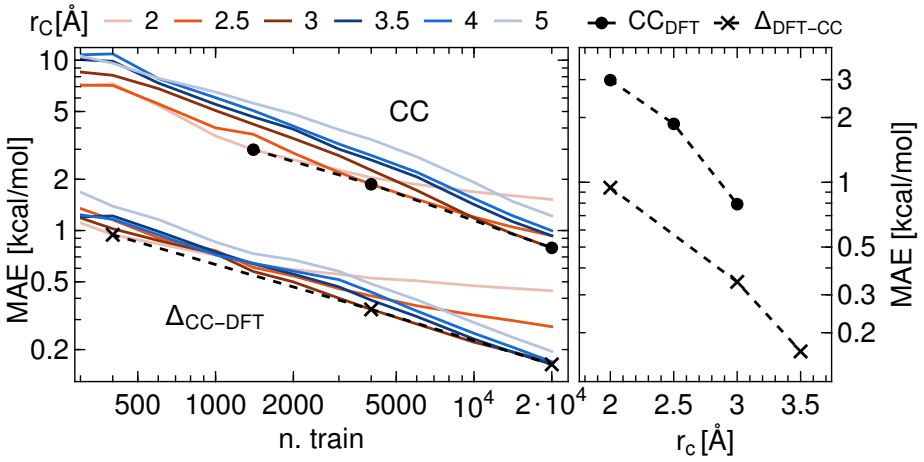

figure S4. Optimal range of interactions for learning GDB9 CC and $\Delta_{\text{CC-DFT}}$ energies. (left) Learning curves for the GDB9 dataset. 20,000 structures were selected by FPS and used for training CC energetics, using the DFT geometries as inputs. The top curves correspond to the error resulting from learning the full CC energy, whereas the lower set of curves correspond to the error that can be achieved using DFT energies as baseline. Tests were performed on 17,000 randomly-selected GDB9 structures, excluding those that were part of the train set. The dashed black curves highlight the envelope of the various training curves, signifying which value of r_C gives the best performance for each training set size. The same points, plotted as a function of r_C (right) give a sense of the energy scale of the interactions that can be modelled with local description over the specified range.

We did not optimize systematically the parameter space, but focused on the cutoff radius r_C that enters the definition of local environments. As shown in Figures **S3** and **S4**, this exercise does not only make it possible to optimize the test error for a given size of the training set, but reveals information on the energy scale associated with different degrees of locality. DFT and CC energies both seem to exhibit a similar trend, with an energy scale of the order of 3 kcal/mol for a very short-range cutoff $r_C = 2$ Å, that decreases below 1 kcal/mol with $r_C = 3$ Å. When considering $\Delta_{\text{CC-DFT}}$, instead, the absolute energy scale is much lower, and one sees that longer-range interactions are crucial: in order to reach an accuracy below 0.2 kcal/mol, $r_C = 3.5$ Å is the best choice for the SOAP environment cutoff.

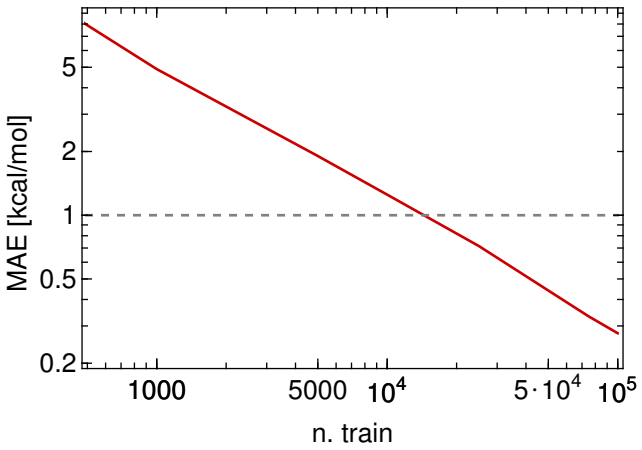

figure S5. Training curves for the prediction of DFT energies using DFT geometries as inputs for the GDB9 dataset. We selected about 33k structures as random to be used as a test set, and then sorted the remaining 100k structures in FPS order, and computed the MAE as a function of the number of inputs included in the training. We used the same kernel parameters as in the main text, and only increased the cutoff distance to 3.5 Å, to be able to capture finer-grained energetics. The figure demonstrates that the SOAP-GAP model is far from having reached its limiting accuracy when using 20k training inputs. For $n_{\text{train}} = 100,000$ the MAE drops below 0.28 kcal/mol.

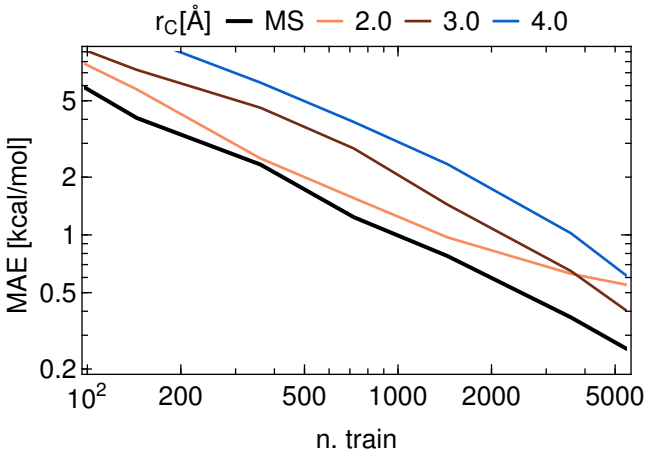

figure S6. Training curves for the prediction of DFT energies using DFT geometries as inputs for the QM7b dataset. Structures are selected in FPS order, and the error is computed on the remainder of the 7,211 configurations. The training curves for different SOAP cutoff length follow a similar trend to what is observed for the GDB9, with a trade-off between completeness of the description, and the extrapolative power for small training set size. The thicker black curve, labelled MS (for multi-scale) uses a compound kernel built by averaging together the three kernels with different cutoff lengths.

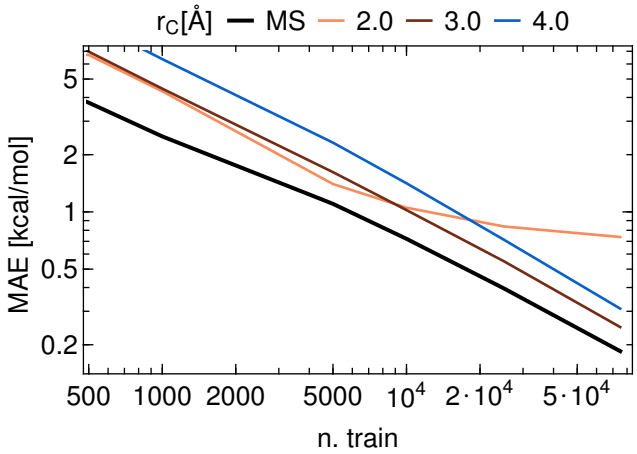

figure S7. Training curves for the prediction of DFT energies using DFT geometries as inputs for the GDB9 dataset. The error is computed on 34,000 randomly-selected structures, the training is performed on structures selected in FPS order from the remaining 100,000 configurations. The training curves for different SOAP cutoff length follow a similar trend to what observed for CC energies, and for QM7b. The thicker black curve, labelled MS (for multi-scale) uses a compound kernel built by averaging together the three kernels with different cutoff lengths.

D. DFT-on-DFT benchmarks

Although in this work we focused on obtaining *useful* predictions, that would allow one to circumvent expensive electronic-structure calculations, most of the benchmarks in recent literature have been performed using DFT-optimized geometries as the input for predicting DFT energetics. In order to compare with other state-of-the art machine-learning models, and to provide an idea of the limiting accuracy and the scope for improvement for the SOAP-GAP model, we have also performed this kind of benchmark calculations. Figure **S5** demonstrates the behavior of the SOAP-GAP model for the GDB9 database when the number of training points is increased above 20,000. One can see that the error is far from saturating, and a MAE below 0.3 kcal/mol can be achieved with the same simple class of kernels we used in the main text, by increasing the training set to contain 100k structures.

We also attempted a preliminary demonstration of the possible directions in which one could improve the performance of SOAP-GAP kernels for a fixed training set size. For these tests we used the smaller QM7b dataset [8], that contains 7,211 molecules with up to 7 N,O,C,Cl,S atoms, with different degrees of H saturation. Our early study applying SOAP descriptors to this system Ref. [20], where we used a considerably more complex non-additive kernel with far from optimal parameter settings, demonstrated 1 kcal/mol MAE with 75% of the data set used for training. With the same training-set size, the present, much simpler, additive SOAP-GAP framework achieves

a MAE of 0.4 kcal/mol with a cutoff of 3 Å. The dependence of the training curves on cutoff radius is similar to what we observed for the GDB9 (Figure **S6**), with a tradeoff between the ultimate attainable accuracy and the extrapolative power for small training set size.

A very simple approach to improve the accuracy of our framework even further entails combining information from different length scales. Within a Bayesian formalism, one can just build a linear combination of different kernels, weighted by a factor that represents the relative contribution to the target property. Such a multi-scale kernel (specifically, one built as $k_{\rm MS} = (256k_{r_c=2}+16k_{r_c=3}+1k_{r_c=4})/273$) reduces the MAE consistently across training set sizes, reaching a MAE of just 0.26 kcal/mol with a training-set containing 75% of the overall data (Fig. **S6**). The same combination of kernels also enables dramatic improvements in the prediction of DFT energies for GDB9. As shown in Fig. **S7** using a multi-scale kernel combining information from 2, 3, 4 Å makes it possible to reach MAE below 1 kcal/mol with about 5,000 training points, that drops to a minuscule 0.18 kcal/mol by the time the train set contains 75,000 structures. Both the results on GDB9 and on QM7b are considerably better than similar benchmark calculations on these two databases [64, 65].

Another direction in which the SOAP descriptors can be improved involves using a choice other than $\kappa_{\alpha\beta} = \delta_{\alpha\beta}$ in the "alchemical" component of the kernel. $\kappa_{\alpha\beta}$ represents the "overlap" between different elements in the definition of the SOAP kernel, that is

$$k(\mathcal{X},\mathcal{X}') = \int \mathrm{d}R \left| \sum_{\alpha\beta} \kappa_{\alpha\beta} \int \mathrm{d}\mathbf{x} \rho_\alpha(\mathbf{x}) \rho'_\beta(R\mathbf{x}) \right|^2 , \quad (5)$$

where ρ_α and ρ'_β correspond to the densities stemming from the species α and β in the environments $\mathcal{X}$ and $\mathcal{X}'$ respectively (see Ref. [20] for a more thorough discussion). We did not attempt a systematic study of the role of these hyperparameters – that represent correlations between the properties of different elements – but experimented with a definition of the form $\kappa_{\alpha\beta} = e^{-(a_\alpha - a_\beta)^2/2\Delta^2}$, where a_α represent an atomic property. Results are promising: using the first ionization energy for a and $\Delta = 1$ eV we obtained (for DFT-on-DFT QM7b, with the reference 75% training set size, and a SOAP cutoff of 3 Å) a MAE of 0.38 kcal/mol. Using the electron affinity and $\Delta = 1$ eV, we obtained a MAE of 0.34 kcal/mol. Using Pauling electronegativity and $\Delta = 0.5$ we achieved a MAE of 0.33 kcal/mol.

E. Oligopeptides

To test the extrapolation capabilities of the SOAP-GAP model built on the GDB9, we considered a few hundred structures from a database of gas-phase conformers of proteinogenic oligopeptides [25]. We picked in particular 500 local minima for glutammic acid dipeptide (E) and for 184 local minima for aspartic acid dipeptide (D) (containing respectively 14 and 13 non-H atoms), and re-optimized the geometries using exactly the same density-functional protocol as used for the GDB9. We then proceeded to test the performance of the GDB9-trained models in predicting the relative stability of the different conformers. We started from the rather academic exercise of using DFT-optimized geometries to predict DFT energetics. As shown in Fig. **S8**, GDB9-trained model provides predictions with an accuracy comparable to DFT – not only of the absolute stability of the two compounds, but also of the relative stability of different conformers. The model can be improved systematically by including structures from the oligopeptides dataset.

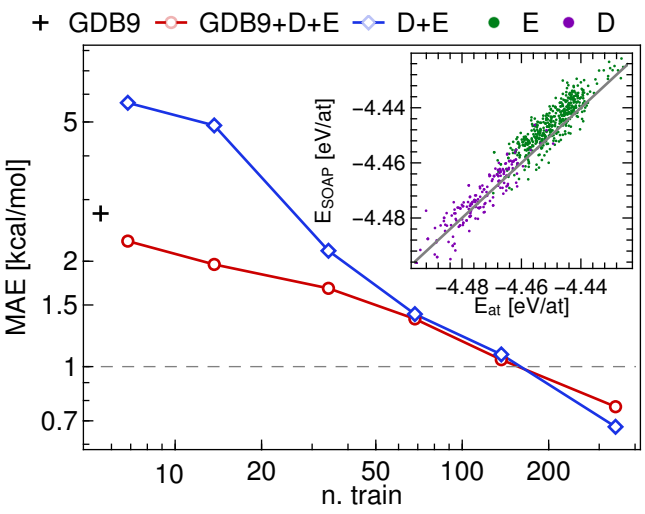

figure S8. Errors in the learning of conformational stability of dipeptides based on GDB9. Training curves for the prediction of DFT energies using DFT geometries as inputs, for a dataset containing a total of 684 configurations of glutamic acid dipeptide (E) and aspartic acid dipeptide (D). The inset shows the correlation between DFT and ML energies as obtained from the model trained on 20,000 FPS-selected structures from the GDB9, which has a MAE of 2.8 kcal/mol – and would already be sufficient for a preliminary screening of candidate conformers. The model can be systematically enhanced by including FPS-selected conformations from the oligopeptide dataset. With about 20% of the structures, both the extendend GDB9 model and a model trained directly on the oligopeptides conformers reaches the 1 kcal/mol milestone.

Figure **S9** shows an analysis of the predictive power of the GDB9-trained model for the DFT-to-CC corrections $\Delta_{\text{DFT-CC}}$. Not only can the SOAP-GAP model correct the large discrepancy between the DFT and the CC atomization energies for the two compounds – which can be largely ascribed to atomic corrections, but it can also provide some degree of correction to the *relative energetics* of different conformers of the two molecules – which is remarkable when one considers that this kind of data

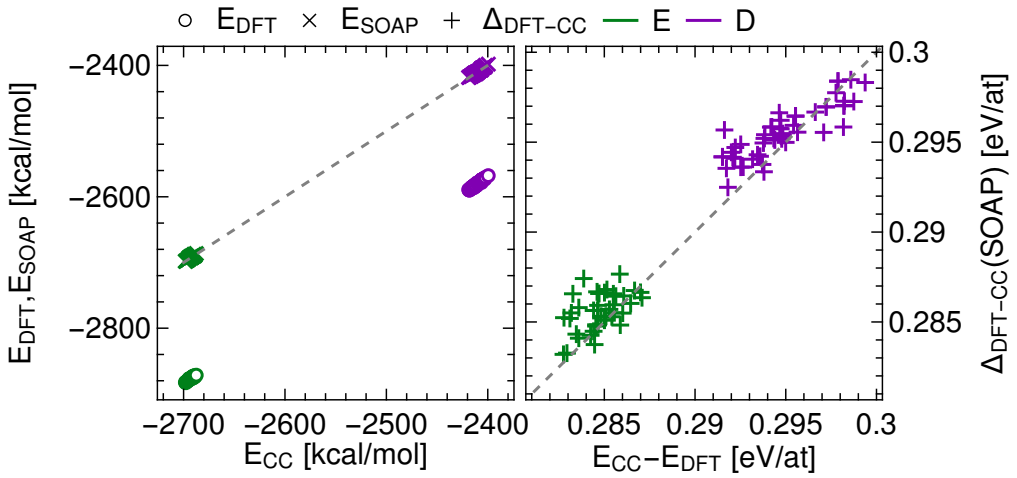

figure S9. Correlation plots for the learning of the energetics of dipeptide configurations, based on GDB9. (left) Correlation between DFT and CC atomization energies for 41 conformers of glutammic acid dipeptide (E) and 52 conformers of aspartic acid dipeptide (D). Disks correspond to the actual DFT and CC energies, crosses correspond to DFT energies corrected with the $\Delta_{\text{DFT-CC}}$ term obtained by the GDB9-trained model. (right) Correlation between the actual difference $E_{\text{CC}} - E_{\text{DFT}}$, and the model prediction.

is not explicitly included in the GDB9.

F. Glucose

As shown in Fig. **S8**, when focusing on a restricted set of compounds, it can be sufficient to use just a handful of training configurations to obtain energy predictions on par with the most accurate electronic structure methods. We considered a set of 208 conformers of glucose, including both closed and open-chain configurations [26]. We use the same SOAP kernel parameters as for the GDB9, and train the model on 20 structures, selected by FPS, using the remaining 188 for validation. As discussed in the main text, this brings the typical error in the energy of conformers relative to benchmark, complete-basis-set CCSD(T) values to less than 0.2-0.4 kcal/mol when using DFT as a baseline, corresponding to a reduction between 50 and 80% of the MAE, relative to the intrinsic discrepancy between the two methods.

section 4. Ligand Classification and Visualisation

The classification of ligands from the DUD-E into actives and inactives was performed with a Kernel-Support-Vector-Machine with a 1-norm penalty factor $C = 1.0$. The decision function for a test structure B is then

$$z_B = \sum_A \alpha_A^* y_A K(A,B) + \beta^*, \tag{6}$$

where $y_A \in -1, +1$ is the class label of a structure A from the training set. The predicted class for B is $\hat{y}_B = \text{sign}(z_B)$. β^* determines the decision threshold, and the coefficients α_A^* are computed based on the optimisation problem (in its dual formulation):

Maximize

$$\sum_A \alpha_A - \frac{1}{2} \sum_{A,A'} y_A \alpha_A K(A,A') y_{A'} \alpha_{A'}, \tag{7}$$

subject to

$$\sum_A y_A \alpha_A = 0, \quad 0 \le \alpha_A \le C. \tag{8}$$

The kernel $K(A,B)$ is chosen as either an average-kernel or "best-match" SOAP (MATCH, in practice a REMatch kernel with $\gamma = 0.01$ [20]). The training is performed on sets of compounds comprising the same number of actives and inactives (decoys), thus automatically assigning equal weight to both classes. SOAP descriptors were generated with `soapxx` software[68] and the following parameters.

```
 "soap-atom": {
"spectrum.global": false,
"spectrum.gradients": false,
"spectrum.2l1_norm": false,
"radialbasis.type" : "gaussian",
"radialbasis.mode" : "adaptive",
"radialbasis.N" : 9,
"radialbasis.sigma": 0.5,
"radialcutoff.Rc": 3.5,
"radialcutoff.Rc_width": 0.5,
"radialcutoff.type": "heaviside",
"radialcutoff.center_weight": 1.0,
"angularbasis.type": "spherical-harmonic",
"angularbasis.L": 6,
"exclude_centers": [],
"exclude_targets": [],
```

```
"type_list": ["Br", "C", "Cl", "F", "H", "I",
    "N", "O", "P", "S"]
}
```

For MATCH, the contribution $\delta z_{j,B}$ of an individual atomic environment $j \in B$ to z_B was computed by decomposing the decision function via the permutation matrix P_{ij}:

$$\delta z_{j,B} = \sum_A \alpha_A^* y_A \sum_{i \in A} P_{ij} k_{ij}(A,B) + \frac{\beta^*}{\|B\|}. \qquad (9)$$

Here, $k_{ij}(A,B)$ is the SOAP kernel between atomic environments $i \in A$ and $j \in B$. We visualised the atomic contributions by defining a density ("binding field") $\rho_B(\boldsymbol{r}) = \sum_{j \in B} \delta z_{j,B} \mathcal{N}(\boldsymbol{r}_j, \sigma_j)$, made up of atom-centered Gaussians $\mathcal{N}$ of width $\sigma_j = 0.5$ Å. This density is subsequently visualised on an isosurface of the atomic density on which the SOAP descriptor is built.

All the ligand binding predictions and binding field maps are available at `http://www.libatoms.org/dude-soap` and individual PDFs for each ligand can be downloaded from `http://www.libatoms.org/dude-soap/pdf/`.