Efficient Selection of Linearly Independent Atomic Features for Accurate Machine Learning Potentials†

Jun-fan Xia, Yao-long Zhang∗, Bin Jiang∗

Hefei National Laboratory for Physical Science at the Microscale, Department of Chemical Physics, Key Laboratory of Surface and Interface Chemistry and Energy Catalysis of Anhui Higher Education Institutes, University of Science and Technology of China, Hefei 230026, China

(Dated: Received on September 12, 2021; Accepted on October 15, 2021)

Machine learning potentials are promising in atomistic simulations due to their comparable accuracy to first-principles theory but much lower computational cost. However, the reliability, speed, and transferability of atomistic machine learning potentials depend strongly on the way atomic configurations are represented. A wise choice of descriptors used as input for the machine learning program is the key for a successful machine learning representation. Here we develop a simple and efficient strategy to automatically select an optimal set of linearly-independent atomic features out of a large pool of candidates, based on the correlations that are intrinsic to the training data. Through applications to the construction of embedded atom neural network potentials for several benchmark molecules with less redundant linearly-independent embedded density descriptors, we demonstrate the efficiency and accuracy of this new strategy. The proposed algorithm can greatly simplify the initial selection of atomic features and vastly improve the performance of the atomistic machine learning potentials.

Key words: Linearly independent, Feature selection, Atomic descriptor, Machine learning, Embedded atom neural network

I. INTRODUCTION

In recent years, various machine learning methods have been introduced in constructing accurate potential energy surface (PESs) from first-principles data [1–14]. To achieve efficient and accurate training of such data, the representation of the molecular structures which is used as the input of the machine learning program should encode the intrinsic symmetry information of the system so as to warrant the continuity and invariance of the PES with respect to overall rotation, translation, permutation of identical atoms. This symmetrization can be systematically done by using permutation invariant polynomials (PIPs) [15] or fundamental invariants (FIs) [3] in terms of internuclear distances as the input of neural networks (NNs) [16, 17] and Gaussian process regression (GPR) [18] in relatively low-dimensional systems. For a given system, a minimum set of PIPs or FIs that are linearly independent of each other can be well determined from invariant theory [3]. However, the order and number of PIPs and FIs increase drastically with the system size, making them computationally infeasible for high-dimensional systems.

An alternative way is expressing the total energy as the sum of atomic energies dependent on the local...
cally atomic structures [7], which are then often characterized by the so-called atomic descriptors (or fingerprints/features) based on various atom-centered many-body expressions [19]. Some of these atomic descriptors, such as the atom-centered symmetry functions (ACSFs) [20] used in the Behler-Parrinello neural network (BPNN) [7] approach, consist of a set of many-body functions with different parameters to describe different correlations between atoms within the chemical environment. These functions may be linearly dependent and contain redundant correlation information. Other descriptors, such as the smooth overlap of atomic positions (SOAP) [19], although can be derived in a systematic way, they often come with tens of thousands of elements that are computationally expensive if no contraction is imposed [21].

Theoretically, increasing the size of atomic descriptors enables a higher representability for atomic structures, but at the expense of a higher computational cost. An optimal choice of important features and/or their parameters is therefore highly desirable. Some empirical rules to determine parameters of atomic descriptors, e.g. the widths and centers of radial Gaussian functions, have been summerized [22, 23]. The intrinsic parameters have been also adapted by a genetic algorithm [22] or a Monte-Carlo algorithm [24]. More recently, Ceriotti and coworkers [25] have discussed several protocols to select the atomic fingerprints out of a large pool of candidates based on the intrinsic correlations (or similarities) of the training data, including the CUR decomposition [26], farthest point sampling (FPS), and Pearson correlation (PC) algorithms, among which the CUR decomposition method was found to perform best. This CUR-based feature contraction scheme needs no supervised learning, but has to instead repeat singular value decomposition (SVD) to score each feature, which will become increasingly time-consuming and memory-demanding as the size of the pool of candidates increases. In addition, the remaining features are orthogonalized before the next SVD relative to the selected features with the highest score to avoid selecting multiple nearly identical features [25]. This procedure, however, may disorder the correspondence between the selected and initial features, since the orthogonalization already reconstructs the feature space. It has been later found that these unsupervised approaches can be improved by incorporating an explicit estimation of the performance of the subset with supervised regression [27]. Goedecker and coworkers have alternatively developed an algorithm to find the largest simplex in the feature space to compress the dimension of the feature vector, which was found to work better than the CUR decomposition [28].

Indeed, such algorithms are designed to select a subset of features that are most linearly-independent. Alternatively, a linear least-square algorithm has been proposed to remove redundant terms so as to find the high-order of fundamental invariants (FIs), a molecular descriptor in the FI-NN method [29]. In this work, following this idea, we propose a simple and efficient way to select atomic descriptors of minimal linear correlation. In Sec. II, we will introduce the embedded density descriptors [23] used here and the algorithm to efficiently select a limited number of linearly-independent terms out of a large pool. In Sec. III, we will present numerical examples of applying this algorithm to fit several benchmark molecules with different sizes and show the improvement over our previous results based on empirically chosen features. We will then conclude this work in Sec. IV.

II. METHOD

A. Embedded density descriptors

To illustrate the feature selection scheme, we choose to work with our recently proposed embedded atom neural network (EANN) approach [23]. This EANN approach takes advantage of the same energy decomposition as in BPNN, namely $E = \sum_{i=1}^{N} E_i$ for an $N$-atom system, where $E_i$ is the atomic energy for atom $i$ given by an atomic NN. Instead of ACSFs, the physically-inspired embedded density descriptors (EDDs) are used as the input vector of the atomic NN mapping the local atomic structure to potential energy. In particular, an embedded density feature ($\rho_i$) encodes a specific electron density contribution from surrounding atoms at the position of the central atom $i$, reflecting the correlations between these atoms. For simplicity, $\rho_i$ can be expressed by the square of the linear combination of atomic orbitals of neighbor atoms,

$$\rho_i = \sum_{l_x,l_y,l_z} L_{l_x l_y l_z} \left[ \sum_{j \neq i} c_j \phi(r_{ij}) f_c(r_{ij}) \right]^2$$  \hspace{1cm} (1)

where $L_{l_x l_y l_z}$ is the partition of atomic orbitals in a local coordinate system.
where \( N_c \) is the number of neighboring atoms of the central atom \( i \) within a cutoff radius \( r_c \) and a cutoff function \( f_c(r_{ij}) \) warrants that the interaction smoothly decays to zero at \( r_c \). Here, a natural choice of atomic orbital is the Gaussian-type orbital (GTO),

\[
\varphi(r_{ij}) = x^a y^b z^c \exp(-\alpha|\vec{r}_{ij} - \vec{r}_a|^2)
\]

where \( r_{ij} = r_i - r_j \) with \( r_i = (x_i, y_i, z_i) \) and \( r_j = (x_j, y_j, z_j) \) being the Cartesian coordinate vector of the central atom \( i \) and a nearby atom \( j \), \( r = |r_{ij}| \) is the distance between them, \( l_z \) (\( l_y \) or \( l_x \)) is the angular momentum projection on \( x \) (\( y \) or \( z \)) axis, and their sum \( L = l_x + l_y + l_z \) is the total angular momentum, \( \alpha \) and \( r_s \) are parameters that determine the center and width of \( \varphi(r_{ij}) \), namely the radial distributions of the atomic orbitals. It is important to realize that \( c_j \) is the linear expansion coefficient of atomic orbital, which is optimized in the course of training and will not be selected [23]. These EDDs defined in Eq.(1) can be rewritten as a series of radial and angular (when \( L > 0 \)) features [30], preserving the invariance with respect to translation, rotation and permutation. Compared with ACSFs, EDDs incorporate three-body interactions in an implicit way, thus achieving a linear scaling with respect to \( N_c \). This advantage makes EANN much more efficient than the traditional BPNN [23, 31]. There have been successful applications of EANN and its variants [31, 32] across molecules [23], molecule-surface systems [33–37], as well as condensed phase systems [31], and even for learning tensorial properties [38–40].

**B. Selection of linearly-independent atomic features**

Generally speaking, given the fewer number of parameters, choosing the proper parameters for EDDs is relatively easier than those for conventional ACSFs. A reasonable choice of EDDs can be generated with an even distribution of \( r_s \) in the interval of \([0, r_c]\) associated with a fixed \( \alpha \) for all angular momenta \((L)\) [23], which should cover the whole chemical environment within cutoff radius. To increase the fingerprint resolution, one has to increase the number of \( r_s \) grids. This leads to not only an obviously higher computational cost but also possible overfitting issues as many of the EDDs can be strongly-correlated. But this empirical rule allows us to prepare a primitive pool with a fine grid of parameters spanning over the feature space. Concretely, we can specify \( r_s \) values by this equation for a given \( r_c \) and each \( L \),

\[
r_{s,n} = n \frac{r_c}{N_s} = n \Delta r_s \quad (n = 0, 1, \ldots, N_s - 1)
\]

This equation defines \( N_s \) evenly spaced \( r_s \) grids within \( r_c \). The corresponding parameter \( \alpha \) controlling the width of GTOs is related to the spacing \( \Delta r_s \) by [23],

\[
\alpha = \frac{\beta}{(\Delta r_s)^2}
\]

in which \( \beta \) is an adjustable constant and its typical value is 0.2. To avoid selecting too narrow GTOs that are difficult to correlate any atoms in the configuration space, the higher limit of \( \alpha \) is restricted to 0.6. This procedure generates a set of parameters of atomic descriptors to be selected for one element. While for multi-component systems, the pool of candidate atomic descriptors have to be generated by combining all possible parameters of all elements in the system. This will lead to an exponential increase of the size of feature space and quickly render the CUR-based algorithm very expensive. We note that the trial pool of parameters of ACSFs in Ref.[25] for the CUR decomposition was also generated according to a similar empirical rule.

With \( N_f \) atomic features initialized by this strategy and \( M_a \) atomic structures in the training set, we can build up an initial atomic feature matrix (AFM) in \( M_a \times N_f \) dimension,

\[
\rho = \begin{pmatrix}
\rho_{1,1} & \rho_{1,2} & \cdots & \rho_{1,N_f} \\
\rho_{2,1} & \rho_{2,2} & \cdots & \rho_{2,N_f} \\
\vdots & \vdots & \ddots & \vdots \\
\rho_{M_a,1} & \rho_{M_a,2} & \cdots & \rho_{M_a,N_f}
\end{pmatrix}
\]

where each matrix element corresponds to the value of an atomic feature computed from an atomic environment, each column \( \{\rho_1\} \) thus represents the vector of a candidate atomic feature component projected on different centric atoms, while each row represents the feature vector associated with a specific centric atom, which will serve as the input of the corresponding atomic NN.

Obviously, the initial AFM will involve many linearly-correlated features, which is redundant for the description of atomic structure. Our purpose is to find an optimal \( M_a \times N_f' \) AFM of \( \rho' \) with linearly-independent atomic features, where \( N_f' = N_f \), which still carries suf-
ficient information of the same feature space. This is essentially a dimensionality reduction problem that has been handled by CUR decomposition in Ref.[25]. Alternatively, we solve this problem by picking out those features that are most difficultly expressed by linear combination of others. According to linear algebra, one vector (say $\mathbf{p}_k$) is linearly-correlated with other vectors (say $\{\mathbf{p}_{k'}\}, k \neq k'$), if it can be linearly expressed by some others, namely,

$$\mathbf{p}_k = \sum_{k \neq k'} \gamma_{k'} \mathbf{p}_{k'} \quad (6)$$

This means that a linear regression to $\mathbf{p}_k$ in terms of $\{\mathbf{p}_{k'}\}$ will lead to zero error and determine the coefficients $\gamma$. This motivates us to propose a simple scheme illustrated in FIG. 1. At the beginning, we use $\mathbf{p}'$ consisting of only a random trial column vector to linearly fit other columns $\mathbf{p}_k$ in the initial AFM. The normalized error of the linear least-squares ($\varepsilon_k$) is an indicator of linear independence between atomic features, which is defined as,

$$\varepsilon_k = \frac{\|\mathbf{p}_k - \mathbf{p}' \gamma\|}{\|\mathbf{p}_k\|} \quad (7)$$

In practice, the largest $\varepsilon_k$ suggests a largest linear independence between $\mathbf{p}_k$ and $\mathbf{p}'$ so that the former should be incorporated into $\mathbf{p}'$. Meanwhile, the column vectors with very small errors indicate their linear-dependence with $\mathbf{p}'$ and will be removed from the original AFM. Both $\mathbf{p}'$ and AFM are then updated, followed by the next iteration. This process will be repeated until all fitting errors are below a certain tolerance or the size of the AFM reduces to zero, eventually yielding the optimized feature subset out of the whole set. Note that individual column vectors are normalized separately to eliminate the influence of magnitude of feature values on linear fitting.

It is worth noting that a subset of linearly-independent features $\mathbf{p}'$ can play effectively the same role as the original $\mathbf{p}$ in terms of the description of the atomic structure. This is equivalent to adding an extra linear transformation from the original input feature vector $\mathbf{p}$ to $\mathbf{p}'$ before feeding into the atomic NNs, whose coefficients are readily determined by a linear fitting by the subset to the whole set. These coefficients will be incorporated into the weights in the first hidden layer of NNs during training, thus making no sacrifice of fitting accuracy in principle. Since our algorithm relies only on linear fitting, the memory and computational costs will be much smaller than those based on SVD. We estimate at each iteration that our algorithm scaling as $\sim O(MnN_f)$, which is much faster than the CUR-based algorithm in Ref.[25] scaling as $\sim O(\min\{Mn^2, M^2n_f\})$. Moreover, since the linear fitting to each candidate feature can be independently done, a too large AFM can be divided in its column dimension into several smaller sub-matrices, allowing for more facile fittings and a much lower scaling (in principle as low as $\sim O(Mn)$). While the AFM cannot be decomposed in the CUR method, making it much less competitive for systems with complex atomic structures. Note that in our approach, $N_f$ will decrease drastically as these features with low fitting errors are removed, further greatly accelerating the selection procedure and reducing the number of iterations, as shown below.

III. RESULTS AND DISCUSSION

Given the proposed unsupervised learning algorithm, it is interesting to check whether it is able to wisely select a number of features that perform well in the subsequent supervised learning (here NN training) process. Taking the aspirin molecule as an example, with a fixed cutoff $r_c=6.2$ Å, we generate a dense grid of fea-
features with $N_s = 1, 2, 3, \ldots$ [20], and remove those with $\alpha > 0.6$, resulting in 51 candidate features for each element and $N_f = 132651$. One thousand configurations are randomly extracted from the full dataset, resulting in 21000 atomic structures. This gives rise to a huge AFM, from which the linear fitting algorithm iteratively picks out linearly-independent features. The maximum fitting error $\varepsilon_{\text{max}}$ of the selected feature at each iteration is shown in FIG. 2(a). For comparison, the selected features at each iteration are fed to twenty-five differently initialized EANN models, among which the best root-mean-squared-errors (RMSEs) in energy and atomic forces are shown in FIG. 2(b) and (c). Apparently, the linear fitting error of the feature to be selected (i.e. the one with $\varepsilon_{\text{max}}$) decreases rapidly with the increasingly selected number of features ($N_f'$) and then levels off at a negligible error ($\varepsilon_{\text{max}} \approx 8 \times 10^{-3}$). This result clearly indicates that other hundreds of thousands of features are largely linearly-expressed by these dozens of linearly-independent ones with a fast convergence rate. Very similar convergence curves are observed for the EANN predictions in both energy and atomic forces, which is suggestive that the proposed feature selection algorithm is able to find an optimal number of features to converge the fitting quality of EANN potentials with no need to perform multiple time-consuming NN trainings.

To further validate the new approach, we combine it with learning EANN potentials of several small organic molecules including aspirin, ethanol, malonaldehyde, naphthalene, salicylic acid, toluene and uracil, for which benchmark \textit{ab initio} data have been available (http://quantum-machine.org/) and the details of data generation have been described elsewhere [41]. These molecules have up to 21 atoms and C, H, O, N elements, which have been widely used to test the accuracy of a given machine learning model. These benchmark energies and force data have been generated by \textit{ab initio} molecular dynamics (AIMD) simulations at the density functional theory level. A concise dataset with 1000 molecular configurations and a more extensive dataset with 20000 molecular configurations are used for training, respectively, in order to compare the data efficiency of various machine learning models. For feature selection, the initial AFM is generated as discussed above. The linear fitting error tolerance for terminating the selection procedure ($\varepsilon_{\text{max}}$) is set to at $\sim 8 \times 10^{-3}$, which is found sufficient to select the most important features that converge the EANN training errors for all systems. For EANN training, 2 hidden layers of atomic NNs with 40 and 60 neurons in each layer are used for all molecules. More computational details are listed in Tables I and II.

Next, we first compare in Table III the mean absolute errors (MAEs) of energy and force predictions for 50000 configurations given by various machine learning potentials, in which only 1000 structures are used for training. In this data-insufficient regime, the EANN potentials using EDDs handcrafted by the empirical rule (referred as EANN*) reported in our previous work [23] have been found to be generally more accurate with the well-established SchNet [12] potentials, but slightly less accurate than the gradient domain machine learning (GDML) [41] models in particular cases, e.g. for salicylic acid and uracil. This is often explained by the more data-efficient nature of the kernel-based GDML approach [42]. However, with the well-selected linearly-independent features, the prediction errors of EANN potentials (referred as EANN#) significantly de-
TABLE I Numerical details for feature selection of the organic molecules.

| System   | Chemical formula | \(N_f\) | \(N'_f\) | \(\varepsilon_{\text{max}}\times10^3\) |
|----------|------------------|---------|---------|-----------------|
| Aspirin  | C\(_9\)H\(_8\)O\(_4\) | 51\times51\times51 | 24 | 8.4 |
| Ethanol  | C\(_2\)H\(_6\)O | 51\times51\times51 | 24 | 3.1 |
| Malonaldehyde | C\(_2\)H\(_4\)O\(_2\) | 51\times51\times51 | 24 | 4.0 |
| Naphthalene | C\(_{10}\)H\(_8\) | 51\times51 | 24 | 0.8 |
| Salicylic acid | C\(_2\)H\(_4\)O\(_3\) | 51\times51\times51 | 24 | 6.0 |
| Toluene  | C\(_7\)H\(_8\) | 51\times51 | 24 | 2.0 |
| Uracil   | C\(_4\)H\(_2\)N\(_2\)O\(_2\) | 51\times51\times51\times51 | 24 | 5.3 |

TABLE II Comparison of the total number of parameters of SchNet, DeepPot-SE, EANN\(^*\), and EANN\(#\) models for fitting these organic molecules. Bold numbers correspond to the lowest values.

| System   | SchNet | DeepPot-SE | EANN\(^*\) | EANN\(#\) |
|----------|--------|------------|-------------|-----------|
| Aspirin  | 109024 | 131283     | 40779       | 20403     |
| Ethanol  | 109024 | 131283     | 37791       | 20403     |
| Malonaldehyde | 109024 | 87522      | 33290       | 13402     |
| Naphthalene | 109024 | 131283     | 37791       | 20403     |
| Salicylic acid | 109024 | 87522      | 25190       | 13402     |
| Toluene  | 109024 | 175044     | 38356       | 27604     |
| Uracil   | 109024 | 38356      |             |           |

TABLE III Prediction MAEs by various machine learning potentials for the organic molecules with 1000 points. Bold numbers correspond to the lowest values.

| System   | Energies/meV | Atomic forces/(meV/Å) |
|----------|--------------|-----------------------|
|          | SchNet | GDML | EANN\(^*\) | EANN\(#\) | SchNet | GDML | EANN\(^*\) | EANN\(#\) |
| Aspirin  | 16.0 | 11.7 | 14.1 | **10.0** | 58.5 | 42.9 | 43.0 | **30.1** |
| Ethanol  | 3.5 | 6.5 | 4.4 | **2.4** | 16.9 | 34.3 | 20.3 | **7.3** |
| Malonaldehyde | 5.6 | 6.9 | 5.9 | **3.9** | 28.6 | 34.7 | 26.9 | **14.5** |
| Naphthalene | 6.9 | 5.2 | 5.0 | **4.4** | 25.2 | 10.0 | 11.5 | **6.6** |
| Salicylic acid | 8.7 | 5.2 | 6.1 | **4.9** | 36.9 | 12.1 | 22.3 | **11.6** |
| Toluene  | 5.2 | 5.2 | 4.8 | **3.9** | 24.7 | 18.6 | 16.6 | **7.4** |
| Uracil   | 6.1 | 4.8 | 4.8 | **4.4** | 24.3 | 10.4 | 15.3 | **7.6** |

crease by 8.3%–45.5% in energy and 30.0%–64.0% in atomic forces, making them considerably more accurate than other competing machine learning models for all molecules. This suggests that a well-optimized EANN model can be as data-efficient as a kernel-based model.

Increasing the size of the training set enhances the prediction accuracy of all machine learning models and a new comparison is made in Table IV. Note that the GDML model is unavailable due to its poor scaling with respect to the data size. In addition, other deep learning-based models including SchNet [12] and DeepPot-SE [43] are based on 50000 data points with much deeper architectures and more parameters, while our EANN models are trained with a smaller number (15000–20000) of data points and NN parameters. In this data-sufficient regime, the high flexibility of deep learning becomes important. Consequently, the EANN\(^*\) models with empirically-chosen descriptors [23] in general perform less well than SchNet and DeepPot-SE models taking advantage of their self-adjustable descriptors during the learning process, especially for aspirin, naphthalene and toluene. Impressively, however, the optimal selection of atomic features substantially improves the prediction accuracy and the resultant EANN\(#\) potentials turn out to be more accurate than the SchNet and DeepPot-SE ones in most cases. Furthermore, the number of EDDs is greatly reduced so that the total number of parameters in these EANN\(#\) models...
TABLE IV Prediction MAEs by various machine learning potentials for the organic molecules with 15000−20000 points for EANN models, 50000 points for SchNet and DeepPot-SE models. Energies and atomic forces are given in meV and meV/Å, denoted by the first and second numbers in each column. Bold numbers correspond to lowest values.

| System        | Points (15000−20000) | Points (50000) |
|---------------|----------------------|----------------|
|               | EANN*               | EANN#          | SchNet | DeepPot-SE |
| Aspirin       | 6.5, 18.9           | 5.7, 13.8      | 5.2, 14.3 | 6.7, 12.1 |
| Ethanol       | 2.1, 4.0            | 2.0, 2.1       | 2.2, 2.2 | 2.2, 3.1 |
| Malonaldehyde | 3.1, 5.1            | 2.9, 3.3       | 3.5, 3.5 | 3.3, 4.4 |
| Naphthalene   | 4.4, 8.8            | 4.0, 3.8       | 4.8, 4.8 | 5.2, 5.5 |
| Salicylic acid| 4.5, 8.2            | 4.1, 4.5       | 4.3, 8.2 | 5.0, 6.6 |
| Toluene       | 4.3, 9.1            | 3.7, 3.6       | 3.9, 3.9 | 4.4, 5.8 |
| Uracil        | 4.3, 3.6            | 4.1, 2.0       | 4.3, 4.8 | 4.7, 2.8 |

FIG. 3 Comparison of radial distribution functions (RDFs) of all interatomic distances of selected organic molecules obtained by AIMD and MD simulations based on EANN# potentials at 500 K.

potentials is merely about half of that in the EANN* ones and ~1/5 of that in SchNet and DeepPot-SE ones. This result provides encouraging and convincing evidence for the superiority of our feature selection algorithm.

In addition, we run MD simulations based on several EANN# potentials and compute radial distribution functions (RDFs) with those extracted from AIMD trajectories of the training dataset. Specifically, the classical NVT MD simulations are performed at temperature of 500 K up to 200 ps with 0.1 fs time step. The obtained RDFs via AIMD and MD simulations based on EANN# potentials are compared in FIG. 3. The excellent agreement between the AIMD and MD results based on EANN# potentials not only confirms the high accuracy of these EANN# potentials but also indicates that our strategy of selecting linearly-independent atomic descriptors does not suffer from any overfitting issues.

IV. CONCLUSION

An optimal choice of atomic descriptors is known to be essential in obtaining accurate and efficient machine learning representations. It has been shown that in many cases the quality of the structural representation
plays a much more important role than the machine learning regression algorithm itself in determining the accuracy of the PES [44]. In this work, we propose a novel strategy to select a convergent set of linearly-independent atomic descriptors from a large pool. This unsupervised strategy takes the error of linear fitting to a candidate feature by other features as an indicator of the linear dependence between this candidate feature and others. This approach is fully automatic, fast and scalable to different systems. We demonstrate the validity of this algorithm applying it to select an optimal set of EDDs in constructing EANN potentials for a few benchmark organic molecules. It is quite efficient and robust, leading to much more accurate and efficient EANN potentials than previous ones constructed with empirically selected features, as well as other well-established machine learning potentials. It is worthwhile to note that the proposed feature selection approach is of course not limited to the present EDDs and can be easily adapted to other classes of atomic descriptors. This approach can be also extended to screen similar configurations from a large pool of data points or in the mean time of active learning [45, 46], if we select row vectors instead of column vectors in the AFM. Combining feature selection and data selection using this approach will be an interesting direction to explore in the future.

V. ACKNOWLEDGMENTS

This work was supported by CAS Project for Young Scientists in Basic Research (YSBR-005), the National Natural Science Foundation of China (No.22073089 and No.22033007), Anhui Initiative in Quantum Information Technologies (AHY090200), and the Fundamental Research Funds for Central Universities (WK2060000017). We thank Kejie Shao at Dalian Institute of Chemical Physics for some insightful discussion, who inspired us to start this project.

[1] S. Manzhos, R. Dawes, and T. Carrington Jr., Int. J. Quant. Chem. 115, 1012 (2015).
[2] B. Jiang, J. Li, and H. Guo, Int. Rev. Phys. Chem. 35, 479 (2016).
[3] K. Shao, J. Chen, Z. Zhao, and D. H. Zhang, J. Chem. Phys. 145, 071101 (2016).
[4] M. Majumder, A. S. Ndengué, and R. Dawes, Mol. Phys. 114, 1 (2016).
[5] R. V. Krems, Phys. Chem. Chem. Phys. 21, 13392 (2019).
[6] B. Jiang, J. Li, and H. Guo, J. Phys. Chem. Lett. 11, 5120 (2020).
[7] J. Behler and M. Parrinello, Phys. Rev. Lett. 98, 146401 (2007).
[8] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Phys. Rev. Lett. 104, 136403 (2010).
[9] A. P. Thompson, L. P. Swiler, C. R. Trott, S. M. Foiles, and G. J. Tucker, J. Comput. Phys. 285, 316 (2015).
[10] A. V. Shapeev, Multiscale Model. Simul. 14, 1153 (2016).
[11] L. Zhang, J. Han, H. Wang, R. Car, and W. E, Phys. Rev. Lett. 120, 143001 (2018).
[12] K. T. Schütt, H. E. Sauceda, P. J. Kindermans, A. Tkatchenko, and K. R. Müller, J. Chem. Phys. 148, 241722 (2018).
[13] J. Zeng, L. Cao, M. Xu, T. Zhu, and J. Z. H. Zhang, Nat. Commun. 11, 5713 (2020).
[14] P. L. Kang, C. Shang, and Z. P. Liu, Acc. Chem. Res. 53, 2119 (2020).
[15] B. J. Braams and J. M. Bowman, Int. Rev. Phys. Chem. 28, 577 (2009).
[16] J. Li, B. Jiang, and H. Guo, J. Chem. Phys. 139, 204103 (2013).
[17] B. Jiang, J. Li, and H. Guo, J. Chem. Phys. 140, 034112 (2014).
[18] C. Qu, Q. Yu, B. L. Van Hoozen, J. M. Bowman, and R. A. Vargas-Hernández, J. Chem. Theory Comput. 14, 3381 (2018).
[19] A. P. Bartók, R. Kondor, and G. Csányi, Phys. Rev. B 87, 184115 (2013).
[20] J. Behler, J. Chem. Phys. 134, 074106 (2011).
[21] J. Nigam, S. Pozdnyakov, and M. Ceriotti, J. Chem. Phys. 153, 121101 (2020).
[22] M. Gastegger, L. Schwiedrzik, M. Bittermann, F. Berzsenyi, and P. Marquetand, J. Chem. Phys. 148, 241709 (2018).
[23] Y. Zhang, C. Hu, and B. Jiang, J. Phys. Chem. Lett. 10, 4962 (2019).
[24] B. Kolb, B. Zhao, J. Li, B. Jiang, and H. Guo, J. Chem. Phys. 144, 224103 (2016).
[25] G. Imbalzano, A. Anelli, D. Giofré, S. Klees, J. Behler, and M. Ceriotti, J. Chem. Phys. 148, 241730 (2018).
[26] M. W. Mahoney and P. Drineas, Proc. Natl. Acad. Sci. USA 106, 697 (2009).
[27] R. K. Cersoncény, B. A. Helfrecht, E. A. Engel, S. Klavininek, and M. Ceriotti, Mach. Learn.: Sci. Technol. 2, 035038 (2021).
[28] B. Parsaeifard, D. Tomerini, D. S. De, and S. Goedecker, J. Chem. Phys. 153, 214104 (2020).
[29] R. Chen, K. Shao, B. Fu, and D. H. Zhang, J. Chem. Phys. 152, 204307 (2020).
[30] A. Takahashi, A. Seko, and I. Tanaka, Phys. Rev. Mater. 1, 063801 (2017).
[31] Y. Zhang, C. Hu, and B. Jiang, Phys. Chem. Chem. Phys. 23, 1815 (2021).
[32] V. Zaverkin and J. Kästner, J. Chem. Theory Comput. 16, 5410 (2020).
[33] X. Zhou, Y. Zhang, R. Yin, C. Hu, and B. Jiang, Chin. J. Chem. 39, 2917 (2021).
[34] X. Zhou, Y. Zhang, H. Guo, and B. Jiang, Phys. Chem. Chem. Phys. 23, 4376 (2021).
[35] R. Yin and B. Jiang, Phys. Rev. Lett. 126, 156101 (2021).
[36] C. Hu, Y. Zhang, and B. Jiang, J. Phys. Chem. C 124, 23190 (2020).
[37] L. Zhu, Y. Zhang, L. Zhang, X. Zhou, and B. Jiang, Phys. Chem. Chem. Phys. 22, 13958 (2020).
[38] Y. Zhang, S. Ye, J. Zhang, C. Hu, J. Jiang, and B. Jiang, J. Phys. Chem. B 124, 7284 (2020).
[39] Y. Zhang, R. J. Maurer, and B. Jiang, J. Phys. Chem. C 124, 186 (2020).
[40] W. K. Chen, Y. Zhang, B. Jiang, W. H. Fang, and G. Cui, J. Phys. Chem. A 124, 5684 (2020).
[41] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K. R. Müller, Sci. Adv. 3, e1603015 (2017).
[42] O. T. Unke, S. Chmiela, H. E. Sauceda, M. Gastegger, I. Poltavsky, K. T. Schütt, A. Tkatchenko, and K. R. Müller, Chem. Rev. 121, 10142 (2021).
[43] L. Zhang, J. Han, H. Wang, W. A. Saidi, R. Car, and E. Weinan, in Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, Canada: Curran Associates Inc., 4441 (2018).
[44] F. A. Faber, L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley, and O. A. von Lilienfeld, J. Chem. Theory Comput. 13, 5255 (2017).
[45] Q. Lin, Y. Zhang, B. Zhao, and B. Jiang, J. Chem. Phys. 152, 154104 (2020).
[46] Q. Lin, L. Zhang, Y. Zhang, and B. Jiang, J. Chem. Theory Comput. 17, 2691 (2021).