JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

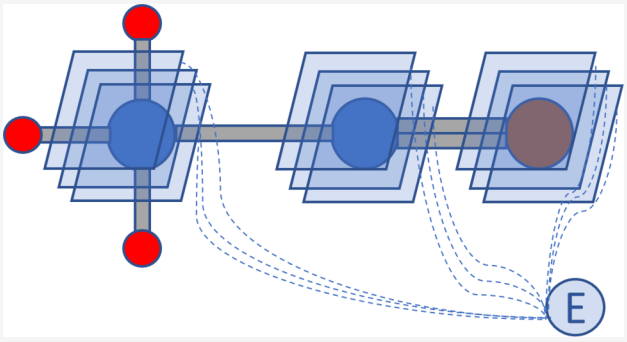
A Multiple Filter Based Neural Network Approach to the Extrapolation of Adsorption Energies on Metal Surfaces for Catalysis Applications

Asif J. Chowdhury, Wenqiang Yang, Kareem E. Abdelfatah, Mehdi Zare, Andreas Heyden,* and Gabriel A. Terejanu*

Cite This: *J. Chem. Theory Comput.* 2020, 16, 1105−1114 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Computational catalyst discovery involves the development of microkinetic reactor models based on estimated parameters determined from density functional theory (DFT). For complex surface chemistries, the number of reaction intermediates can be very large, and the cost of calculating the adsorption energies by DFT for all surface intermediates even for one active site model can become prohibitive. In this paper, we have identified appropriate descriptors and machine learning models that can be used to predict a significant part of these adsorption energies given data on the rest of them. Moreover, our investigations also included the case when the species data used to train the predictive model are of different size relative to the species the model tries to predict—this is an extrapolation in the data space which is typically difficult with regular machine learning models. Due to the relative size of the available data sets, we have attempted to extrapolate from the larger species to the smaller ones in the current work. Here, we have developed a neural network based predictive model that combines an established additive atomic contribution based model with the concepts of a convolutional neural network that, when extrapolating, achieves a statistically significant improvement over the previous models.



1. INTRODUCTION

Computational catalyst discovery typically requires the development of a microkinetic model based on parameters determined by density functional theory (DFT) calculations[1] of all reaction intermediates.[2] To minimize the cost of calculating the energies for each reaction intermediate and transition state on different active site models, linear scaling relations[3−5] have been proposed which use a few easily computable descriptors, such as the carbon atom adsorption energy, on different active site models to generate volcano curves on catalyst activity.[6] However, even the DFT computations for only the intermediate species on a number of surfaces require, for a large reaction network with many intermediates, a significant number of expensive calculations. In this work, our goal was to build a predictive framework that would train on the energies of some of the surface species and predict on the rest, which can significantly reduce the computational overhead when working with a complex microkinetic model with a large number of surface species. In addition to that, we also investigated appropriate predictive models for extrapolation of adsorption energies in terms of the size of the species, i.e, when the training data and the prediction set contain different sized molecules. This is typically challenging because machine learning models, while performing satisfactorily during interpolation (when training and testing set come from the same area of the feature space), do not work as well for extrapolation[7] (when training and testing set come from nonoverlapping regions of the feature space).

In this paper, we have worked with two data sets of adsorption energies, both containing reaction intermediates consisting of carbon, oxygen, and hydrogen atoms. One of them contains 247 larger C4 species, i.e., molecules with at least four carbon atoms and variable numbers of oxygen and hydrogen, obtained from the hydrodeoxygenation of succinic acid on Pt(111). The other contains 29 smaller C2 and C3 species, i.e, molecules made up of two or three carbon atoms along with some oxygen and/or hydrogen, obtained from a reaction network of decarboxylation and decarbonylation reactions of propionic acid on Pt(111).[8] All calculations

Received: October 2, 2019
Published: January 21, 2020

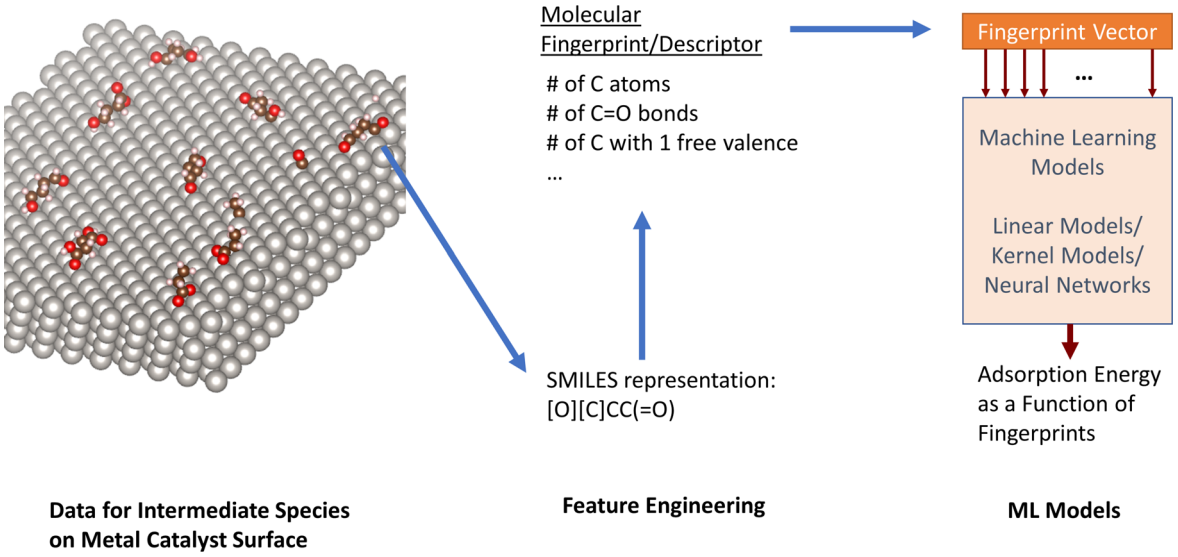


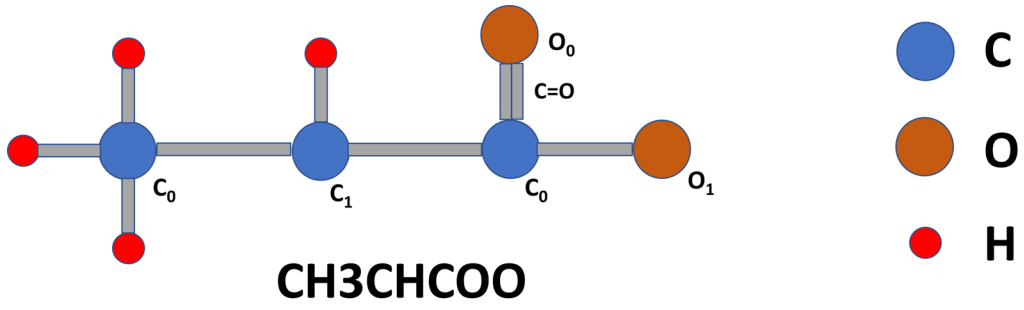
Figure 1. Overview of the application of machine learning for prediction of adsorption energies. Structures of the intermediate species are used to obtain a suitable fingerprint, which is fed to ML models that learn the adsorption energy as a function of the fingerprint vector.

were done using the PBE-D3 functional. Two types of predictive analysis were performed—interpolation on the bigger C4 data set, i.e., training on some of the C4 species and predicting on the rest of them, and extrapolation from the C4 data set to the C2 and C3 data set, i.e., training on the full C4 data and predicting the adsorption energies for the C2 and C3 species. While extrapolation to longer chain molecules is in principle most relevant, we do not possess a C5 data set, and the C2 and C3 data sets are too small to be used for extrapolation to C4 species. Nevertheless, extrapolation from C4 to C2 and C3 is technically as challenging as extrapolation to longer chain molecules, and we expect all of our conclusions to also be valid when extrapolating to longer chain molecules provided these do not contain any chemical fragment that is nonexistent in the smaller training molecules.

Predicting properties of some chemical entity using machine learning[9,10] involves solving two related subproblems—discovery of effective features or descriptors and using a proper machine learning model that, together with the chosen descriptor, works best for the specific task at hand.[11] A high-level workflow for applying machine learning for this process is shown in Figure 1. Here, we are trying to predict the adsorption energies of surface intermediates, and hence the descriptors are essentially some form of molecular fingerprints. Many different kinds of fingerprints or fingerprint generation schemes have been proposed in previous studies— Coulomb matrix[12] and bag-of-bonds[13] using distance measures between the atomic coordinates of the species; atom centered radial or angular symmetry functions;[14−17] noncoordinate based fingerprints that take into account features of a molecule which can be extracted from the chemical formula or SMILES notation;[18−21] generation of fingerprints from molecular graph structure[22,23] where the atoms and the bonds are considered as the nodes and edges of a graph, respectively; and fingerprints corresponding to a target property learned using back-propagation, etc. Fingerprints based on SMILES or graph have the desirable property over the coordinate based descriptors that any DFT or other semiempirical methods need to be applied only on the training data; for the rest of the data for which the adsorption energies are unknown, their molecular notation is all that the predictive model would need to make the predictions. In contrast, the coordinate based methods would need reliable atomic coordinates even for the species on the prediction set which would require some form of expensive calculations—ones we wish to minimize in the first place. Most commonly used machine learning models have been kernel based models such as kernel ridge regression[24] and different neural network based models such as graph convolution,[22,23] recurrent neural network,[25] 3D convolutional neural network which reads the 3D spatial coordinates of the molecule,[26] or additive atomic contribution through atomic subnetworks.[9,14]

In our investigations for interpolation of adsorption energies, we have studied both the coordinate based and SMILES based descriptors along with a variety of machine learning models. Our results indicate that simple molecular descriptors that capture the nearest-neighbor information across the species from the SMILES notation, paired with kernel based models, can perform as good as coordinate based descriptors such as Coulomb matrix or bag-of-bonds with a mean absolute error (MAE) of 0.14 eV. However, for extrapolation, the choice of descriptor is more complex—descriptors based on pairwise or triplet atomic distances such as Coulomb matrix or bag-of-bonds have the disadvantage that they are not size extensible. For data with different sized species, smaller ones have to be padded with zeros that make the learning difficult. In this case, constant sized molecular fingerprints[27] are more suitable. Our results, however, suggest that the predictive errors are still quite high for these descriptors. A different kind of approach, which is atom centered and where each atom's neighborhood information (its pairwise and triplet distances from other atoms) is treated as the atomic fingerprint and fed to a small neural network where these subnetworks from each atom share their weights, and then all of the atoms' contributions are added up to get the final energy, is size extensible. We found this method to work better than the other methods for extrapolation with MAEs of around 0.4 eV. However, the error is still large, and we have sought ways to improve upon this model. One improvement was to include SMILES based atomic fingerprints over the coordinate based ones, and the

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114



C_0	C_1	C_2	C_3	O_0	O_1	C-H	C=O	O-H	$C-O_0$	$C-O_1$
2	1	0	0	1	1	4	1	0	0	1

C_0-C_0	C_0-C_1	C_0-C_2	C_0-C_3	C_1-C_1	C_1-C_2	C_1-C_3	C_2-C_2	C_2-C_3	C_3-C_3
0	2	0	0	0	0	0	0	0	0

Figure 2. Molecular fingerprint for the surface species CH3CHCOO. Here, C_0 denotes a saturated carbon (no free valence). C_1, C_2, and C_3 denote carbon atoms with one, two, and three free valencies, respectively. Similarly, O_0 is a saturated oxygen whereas O_1 is an oxygen atom with one free valence. The fingerprint vector (shown at the bottom of the image) contains the number of different saturated or unsaturated atoms and the number of bonds between them.

second contribution, which helped to get significantly smaller extrapolation errors, was to treat the small atomic subnetworks like a filter of a convolution neural network and use multiple of these filters. This method had extrapolation MAE of 0.23 eV.

2. METHODOLOGY

Molecular fingerprints used in our investigations can be categorized into three classes: first, coordinate based Coulomb matrix and bag-of-bonds that use pairwise distances between the atoms in the molecule to generate the fingerprint; second, flat fingerprints based on the number of different bond counts inside the molecule that can be read from the chemical formula or SMILES notation; third, atom-centered fingerprints that are based on the local neighborhood around an atom which is calculated either by distance measures between the atomic coordinates or by the number of different bond types for the atom that can be read from the molecular notation. Machine learning models that we have used can also be divided into three categories: generalized linear models such as linear regression, ridge regression (which uses L2 regularizer), LASSO (which uses L1 regularizer); kernel based models such as kernel ridge regression (KRR), support vector regression (SVR), Gaussian processes (GP); and artificial neural networks (ANN).

2.1. Coulomb Matrix and Bag-of-Bonds. The Coulomb matrix (CM) method first creates a symmetric matrix where the off-diagonal element $C(i,j)$ is a function of the distance measured between the ith and jth atoms and also their atomic numbers. The diagonal elements are a function of the atomic number of the corresponding atoms. The sorted eigenvalues of the matrix form the molecular fingerprint. The bag-of-bond (BoB) method takes the off-diagonal lower triangle of the symmetric matrix formed in CM and puts the entries corresponding to each atom type pair in a bag, sorts the entries inside each bag, and concatenates the bags to form the fingerprint vector. We have found these methods to typically work well for interpolative predictions among the same sized species. However, for data with variable sized species, one needs to pad the entries of the matrix for smaller species with zeros. This limits their usefulness for size-extrapolation predictions. Detailed methods for building CM and BoB are described in the Supporting Information.

2.2. Flat Molecular Fingerprint. Fingerprints generated from the SMILES notation of the adsorbed species encode the connectivity among the atoms inside the molecule. The encoding can capture the number of different types of atoms or bonds by looking into the nearest neighbors of each atom or up to some specified distance. In our study, we have built a simple scheme, similar to previous work on constant sized descriptors,[27] that looks into the nearest neighbors of the atoms and counts the number of different atom types an atom is connected to and then accumulates the results in a fingerprint vector. The proposed fingerprint is shown in Figure 2. Here, atom types are divided into subclasses by the number of free valencies an atom has, e.g., instead of just looking at how many carbon−carbon bonds are present, the fingerprint captures how many saturated carbon atoms are single bonded to a carbon with one free valence or how many oxygens are double bonded to a saturated carbon atom, and so on. This is a constant sized molecular descriptor as the length of the vector remains the same for smaller or bigger molecules.

As will be discussed later, this method works well for interpolation and works better than CM or BoB for extrapolation, but the extrapolation error was still quite large. Both the flat molecular fingerprint and CM/BoB can be fed to any regular ML model such as linear model, kernel based model, or fully connected feed-forward neural network. In each case, the ML model takes as input the molecular fingerprint vector and outputs the target real value (in our case, adsorption energy). We have also tried the extended

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114

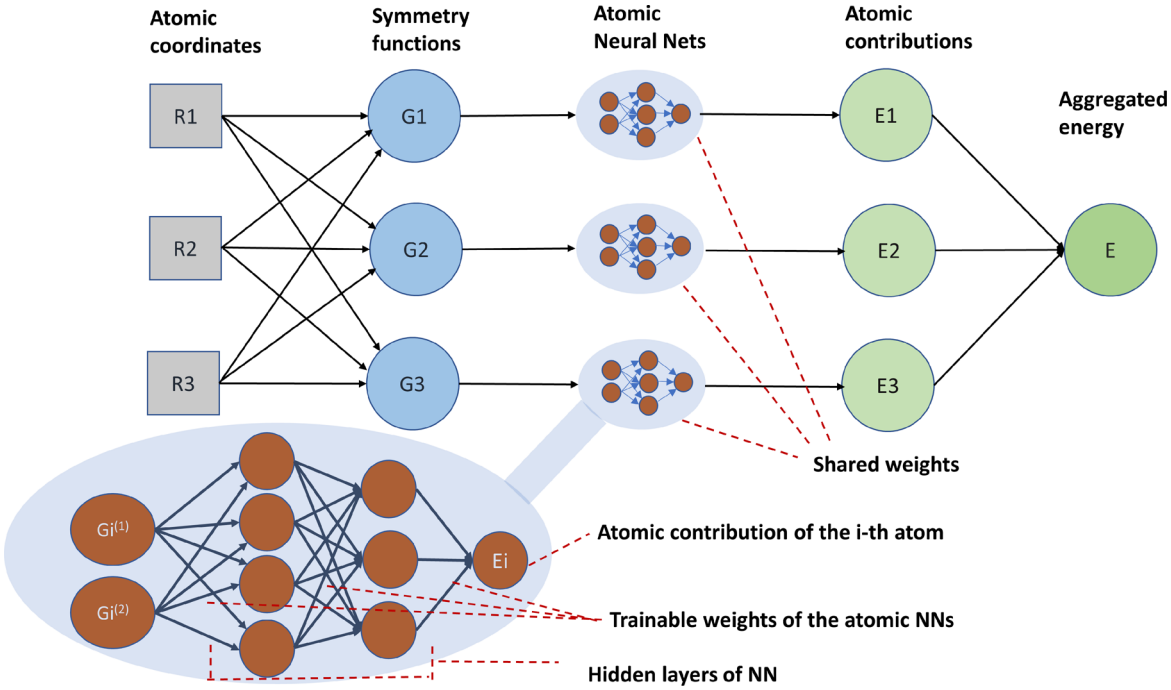


Figure 3. Network for the atomic contribution method. First, symmetry functions are calculated from the atomic coordinates of all the atoms in the molecule. Typically, two symmetry functions are used: one that aggregates the pairwise distance information centered around each atom and the other that combines the angular distance information from a triplet of atoms. Other symmetry functions can be devised, too. Here, G*i* denotes the vector containing the symmetry function values for the *i*th atom. For each atom, its corresponding G vector is fed to a neural network (NN). The networks for all the atoms share weights which make the method work with any ordering of atoms. Each atomic NN learns the energy contribution of the corresponding atom to the total energy of the species. All the atomic contributions are summed to get the predicted energy. The structure of the atomic NN can be adjusted as shown in the bubble at the bottom of the figure.

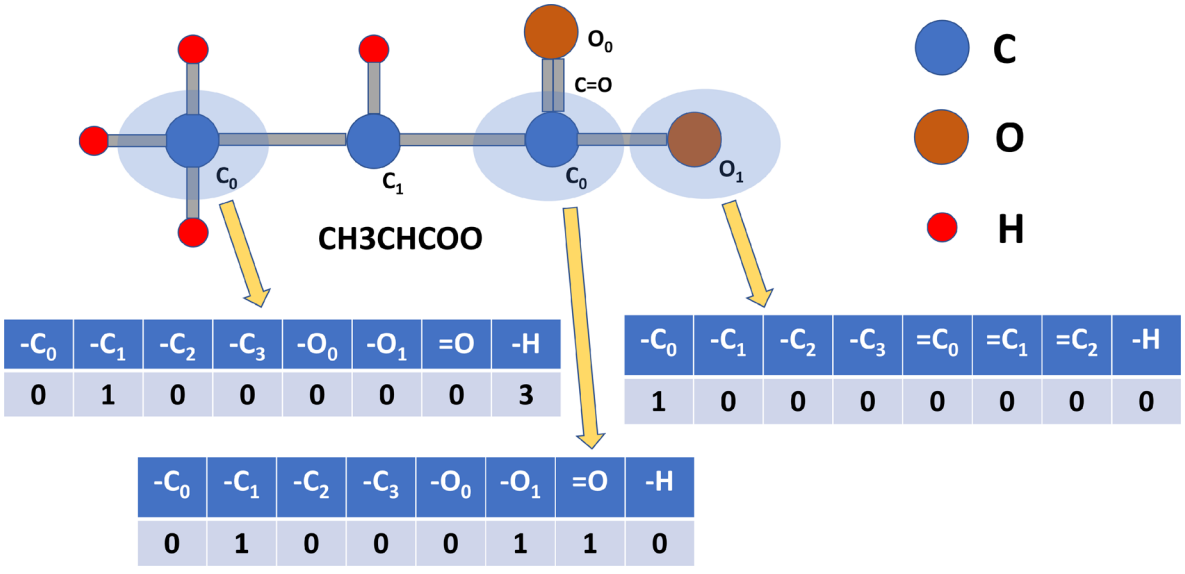


Figure 4. Our proposed noncoordinate based atomic fingerprints for the atomic subnet based method. These fingerprints can be obtained directly from the SMILES notation of the molecule without the need for any atomic coordinates. Three sample fingerprint vectors for two carbon atoms and one oxygen atom are shown. The vectors contain the information about the number of different types of bonds for an atom. For example, the vector item C_1 contains the number of single bonds the current atom has with carbon atoms that contain one free valence. The fifth to seventh positions of the fingerprint vectors have different meaning for carbon and oxygen atoms, e.g, in the seventh position, for carbon, =O denotes how many oxygen to which the current carbon atom is bonded by double bonds, whereas for oxygen, $=C_2$ encodes the number of carbon with two free valencies to which the current oxygen atom is connected by double bonds.

connectivity based fingerprint (ECFP)[18] which produces fixed length vectors from the SMILES notations of the molecules.

2.3. Additive Subnetwork Model. The atomic fingerprint based additive subnetwork model is a size extensible model. The atomic fingerprint originally used[14] for this model was the symmetry functions calculated from the atomic coordinates of all the atoms in the molecule. Two commonly used symmetry functions are one that aggregates the pairwise

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114

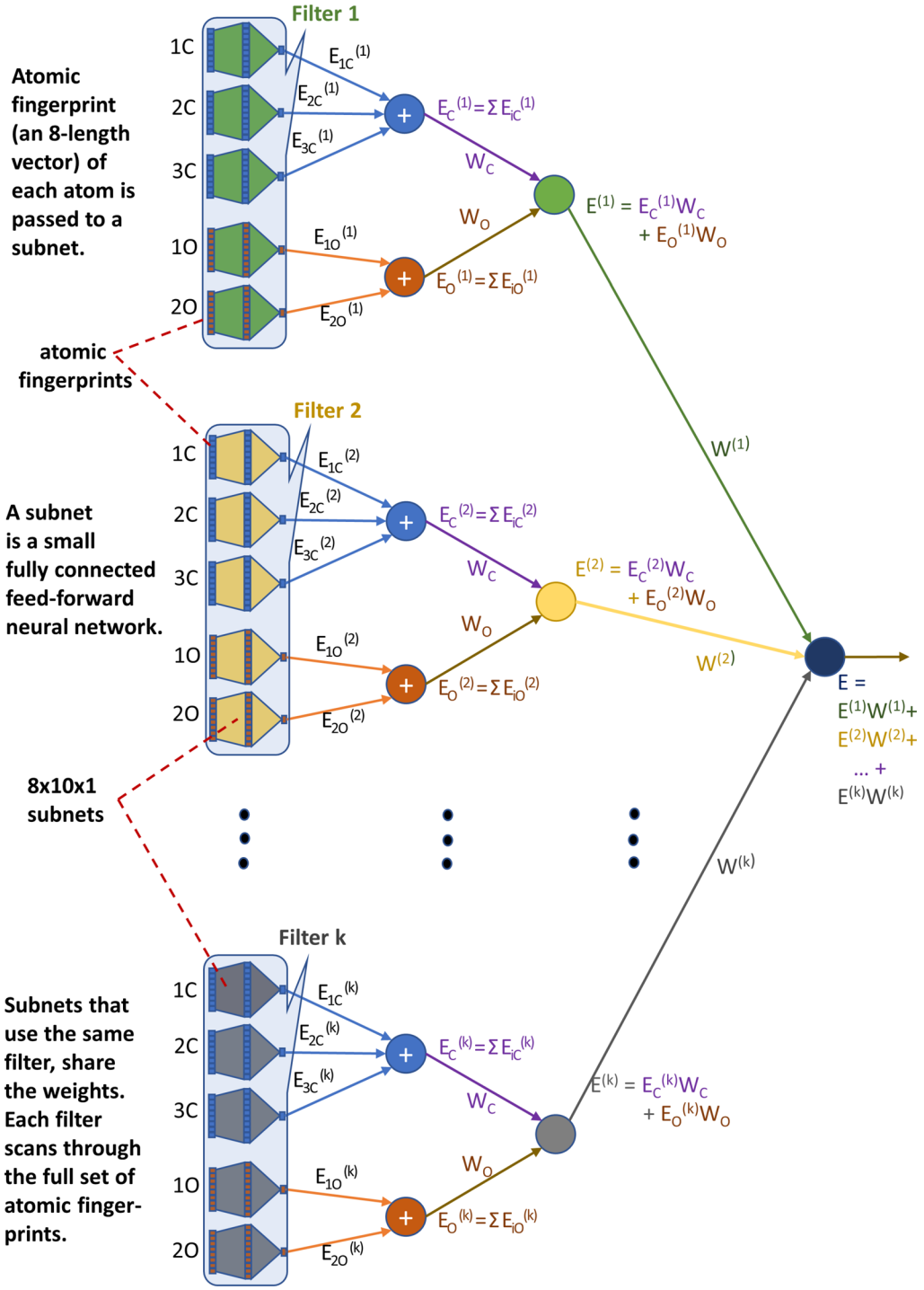


Figure 5. Our proposed model. A species with three carbon and two oxygen atoms is passed through k filters. Each filter is an 8 by 10 by 1 neural network. For each filter, outputs of networks for each of the three C atoms are summed up, and the same is done for the two O atoms. The weighted sum of these two sums is the output for one filter. The final output is the weighted sum of all the filter outputs. The parameters of the network that are learned through back-propagation are $W^{(i)}$s, W_C, W_O, and network weights for each filter.

distance information centered around each atom and the other that combines the angular distance information from a triplet of atoms (equations for these distance measures are given in the Supporting Information). Other symmetry functions can be devised, too. The model is shown in Figure 3.

Fingerprints for each atom are fed to a small neural network. These subnetworks learn the energy contribution of the current atom to the total energy as a function of the fingerprints. Aggregated energy contributions from all the subnetworks yield the final energy. Subnetworks for all the atoms of an atom type share their weights. The weights can also be shared across the atom types. We have found the latter approach to work better for our case. The weight sharing ensures that the model is invariant to the ordering of the atoms. In contrast to other models, this one can only work with neural networks because it gives the flexibility of a hierarchical structure through the use of the back-propagation method to learn the network weights. In order to avoid the

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114

computation of reliable coordinates for the prediction set, we prefer the SMILES based fingerprints. We have developed such an atomic fingerprint, shown in Figure 4, which is similar to the flat molecular fingerprint described above but is centered on an atom and encodes the connectivity information for that atom. We have found this model to work better for extrapolation compared to CM, BoB, or flat fingerprints, but the errors were still quite large, which warranted further improvements to the model.

2.4. Proposed Multiple Filter Based Additive Subnetwork Model. To make the additive subnetwork based model generalize better to an unseen testing data, we propose to treat the shared weights of the atomic subnetworks as a filter in a convolutional neural network (CNN). This type of deep learning model is commonly used for image data where different sets of weights (called convolutional filter) scan through the image patches and learn to detect various basic image features such as edges or corners[28] which are combined in subsequent layers to detect higher level objects such as a face or a car or a digit.[29] These filters are analogous to the local receptive field in biological visual systems.[30,31] In CNN, the filters are usually 2D or 3D matrices of weights. A filter is placed on a patch of the image, and a cross-correlation operation between the filter weights and the input plane pixels is performed—this is the output of that filter for that image patch. Then, the filter is moved to the next adjacent patch (which may or may not overlap with the previous patch). A key observation here is that there is not one but a number of filters that are used because each filter learns different features (through back-propagation).[32] Moving to our problem and the atomic subnetworks, we can think of a subnetwork as a filter in CNN. Since all of the subnetworks share their weights, they can be considered as one filter scanning each of the atoms in the molecule one by one. Unlike CNN, however, in our case the subnetworks compute a nonlinear function of its inputs instead of the cross-correlation, which makes sense since predicting adsorption energies is a regression problem and we want each subnetwork to learn the energy contribution of an atom. Also, unlike CNN, here we do not need multiple layers of filters as our learning objective is to find the individual atom contribution to the total energy. However, the aspect of CNN that can be incorporated in our networks and that can lend the additive subnetworks a better representational ability is to use multiple filters instead of one. Here, we should clarify that using multiple filters does not mean using separate filters for different atom types. Whether we use different shared weights for different atom types or not, by "multiple filters" we are referring to completely separate sets of filters (in each set, there may be one filter if all atom types share the weights or more than one if weights are shared only inside each atom type). In our proposed model, each of the separate set of filters would scan each atom of the molecule, and the weighted sum of all the filtered values should yield the final output energy.

The proposed model is shown in Figure 5. The atomic fingerprints are the 8-length vectors from Figure 4. During the training of the network, at each iteration of the gradient descent, there is a forward pass that starts from the fingerprints at the left of the figure and moves to the right. The gradient of the error in the energy obtained at the right-most node is then back-propagated through the network which makes it learn the appropriate weights to fit the data. Nonlinearity in the computation comes from the nonlinear activation functions used in the hidden layers of the filters. The error function (that the gradient descent tries to optimize) for neural networks is nonconvex[33]—there can be many local minima. This means the output of a neural network is sensitive to the starting points of its weights; starting from different points in the hyperspace can result in different ending locations. Since the weights of each filter are randomly initialized,[34,35] each of them is likely to end up with different weights than others even though all of them are fed the same set of atomic fingerprints, i.e., each of them learns different functions of their inputs, just like each CNN filter learns to detect different image features. Let us assume there are K filters and the functions that they learn are denoted as $f^{(1)}, f^{(2)},..., f^{(K)}$ and the output of the ith filter (after combining outputs of that filter for each atom in the species) is $E^{(i)}$. Then, the overall energy output of the networks, E, is

$$E = \sum_{i=1}^{K} E^{(i)} W^{(i)} \tag{1}$$

where $W^{(i)}$ is the weight of the contribution for the ith filter and

$$E^{(i)} = \sum_{a=1}^{T} E_a^{(i)} W_a \tag{2}$$

where T is the number of atom types in the species that have fingerprints (in our case, it is 2 for C and O; since those two atom types can describe all the bonds inside the species, we have not included fingerprints for H), $E_a^{(i)}$ and W_a are the summed contribution for all the atoms of atom type a when passed through filter i and the weight for that atom type which is shared across the filters, respectively. So

$$E_a^{(i)} = \sum_{n=1}^{N_a} E_{a_n}^{(i)} \tag{3}$$

where $E_{a_n}^{(i)}$ is the output when the nth atom of atom type a is passed through filter i. The last equation means the atomic contributions of all the atoms for an atom type for a filter are directly summed and not weighted (which can be treated like a constant, nonlearnable weight of 1). This ensures that a change in the relative ordering of the atoms (inside the set of atoms of an atom type) does not change the overall result. If the atomic fingerprint for the atom a_n is X_{a_n}, then $E_{a_n}^{(i)}$ is a nonlinear function of X_{a_n}:

$$E_{a_n}^{(i)} = f^{(i)}(X_{a_n}) \tag{4}$$

Here, the fingerprint is passed to the filter which, in our case, is a small fully connected feed-forward neural network (NN). The output of each layer in the NN is computed by multiplying the weight matrix between the current and the previous layer with the output vector of the previous layer and then passing the obtained vector to a nonlinear activation function.[36−38]

3. RESULTS AND DISCUSSION

In our investigations, we have used Coulomb matrix, bag-of-bonds, flat molecular fingerprints (noncoordinate based, calculated from the SMILES), and additive atomic subnetwork models for both interpolation and extrapolation. Key results are shown in Table 1 and Table 2, respectively. The Supporting Information contains full tables of all results. Here, the table for interpolation shows that noncoordinate

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114

Table 1. Interpolation Results[a]

method	ML model	MAE (eV)	SD of AE (eV)
Coulomb matrix	GP	0.230	0.218
bag-of-bonds	KRR	0.139	0.136
bag-of-bonds	ridge	0.219	0.279
ECFP	SVR	0.165	0.179
flat molecular fingerprint (from SMILES)	SVR	0.148	0.129
flat molecular fingerprint (from SMILES)	KRR	0.141	0.122
fllat molecular fingerprint (from SMILES)	ridge	0.196	0.166
additive atomic subnetwork	ANN	0.398	0.202
proposed model (from coordinates, 1 filter)	ANN	0.347	0.259
proposed model (from coordinates, 4 filters)	ANN	0.309	0.231
proposed model (from SMILES, 1 filter)	ANN	0.190	0.164
proposed model (from SMILES, 6 filters)	ANN	0.142	0.120

[a]The methods used: Coulomb matrix, bag-of-bonds, flat molecular fingerprint, and additive atomic subnetwork model (see discussions for details). For the first seven rows, for each of the methods, we ran the following ML models: ridge regression, LASSO, kernel ridge regression (KRR), support vector regression (SVR), Gaussian processes (GP). The rest of the rows used artificial neural networks (ANN). Some of the results from each category are shown. The first and second columns show the descriptor method and the ML model used, respectively. The third column contains the mean absolute error (MAE) of the predicted adsorption energies, and the fourth column presents the standard deviations of the absolute errors. Data for the 247 C4 species were randomly permuted, and 215 were used for training and the rest for testing. The process was repeated 100 times (data being permuted randomly each time) to obtain an unbiased estimate of the MAE.

Table 2. Extrapolation Results[a]

method	ML model	MAE (eV)	SD of AE (eV)
Coulomb matrix	SVR	2.392	1.015
bag-of-bonds	KRR	2.046	0.422
ECFP	SVR	2.961	0.760
flat molecular fingerprint (from SMILES)	KRR	2.342	0.625
additive atomic subnetwork	ANN	0.441	0.214
proposed model (from coordinates, 1 filter)	ANN	0.324	0.212
proposed model (from coordinates, 4 filters)	ANN	0.282	0.190
proposed model (from SMILES, 1 filter)	ANN	0.434	0.314
proposed model (from SMILES, 5 filters)	ANN	0.227	0.143

[a]The methods used: Coulomb matrix, bag-of-bonds, flat molecular fingerprint, and additive atomic subnetwork model (see discussions for details). For the first four rows, for each of the methods, we ran the following ML models: ridge regression, LASSO, kernel ridge regression (KRR), support vector regression (SVR), Gaussian processes (GP). The rest of the rows used artificial neural networks (ANN). Some of the best results for each method are shown. The first and second columns show the descriptor method and the ML model used, respectively. The third column contains the mean absolute error (MAE) of the predicted adsorption energies, and the fourth column presents the standard deviations of the absolute errors. Data of 247 C4 species were used for training, and data of 29 C2 and C3 species were used for testing.

based molecular fingerprints with kernel based ML models perform as well as coordinate based descriptors with the same ML models. The additive atomic subnetwork based on SMILES with multiple filter also worked well. For this model, we also used a coordinate based atomic fingerprint of length 5—aggregated pairwise distance measured from an atom to each of the four atom types involved (carbon, hydrogen, oxygen, and top two layers of metal catalyst surface), plus the triplet distance measure.

For extrapolation, both the Coulomb matrix and bag-of-bonds performed poorly. This is not unexpected since these methods are not size extensible and require padded zeros to make them work for different sized molecules. From this point of view, the flat molecular fingerprint comes as an attractive alternative as it is a constant sized descriptor (size of the molecule does not effect the size of the vector; no zero padding is required). But our results show that it performs no better than CM or BoB for extrapolation. However, the method that we found to be most promising was the additive atomic subnetwork. Since this method adds up the atomic contributions to the total energy, it is naturally size extensible. The initial predictive error obtained using this method (with the length 5 fingerprint discussed above) was approximately 0.4 eV. As a neural network ends up in a different location of its parameter hyperspace on different runs (because of randomly initialized parameters), we ran the model multiple times, and our final result was an ensemble of these runs—for each target species, its predicted adsorption energy was the mean of its predicted values of all the runs. This yielded an extrapolation error of approximately 0.32 eV. The ensemble method was used in all of our following models.

The next step was to replace the coordinate based atomic fingerprints with SMILES based ones (Figure 4). However, only replacing the atomic fingerprints in the additive subnetwork model actually increased the predictive errors to over 0.4 eV. We improved this model by using multiple filters as discussed before (Figure 5). The predictive errors went down significantly as more filters were used. The rate of improvement, however, gradually subsided, and after incorporating a certain number of filters, the predictive errors change very little, as can be seen in Figure 6. We used this multifilter approach with the coordinate based atomic fingerprints as well, and the extrapolation error there went down to 0.28 eV from over 0.32 eV.

Here, we should note that the number of filters is essentially a hyperparameter to our model and needs to be tuned for specific problems. Tuning of hyperparameters for machine learning models is typically done by setting aside a portion of the training set as validation set and choosing the hyperparameters for which the model performs best on the validation set. The chosen model is then used to run on the testing set. We also used this approach. This works well for interpolation problems where the training (which includes the validation set) and testing sets reside in the same region of the parameter space. But in case of extrapolation, this might not work.

Through our investigations, we have seen that for extrapolation, the error on the validation set (which is part of the training set, containing C4 species) went to an approximate minimum value when six filters were used and then remained more or less constant. But the extrapolation

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114

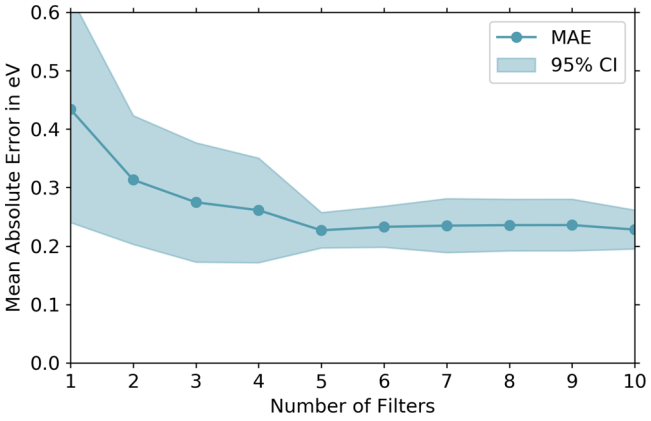


Figure 6. Extrapolation errors decreased sharply with the use of more filters. At one point, however, it reaches a state where adding more filters does not make any significant improvement. The model used here is our proposed model shown in Figure 5, and the atomic fingerprints are the ones shown in Figure 4.

error on the testing set (containing C2 and C3 species), after the network was trained with different numbers of filters, reached minimum with five filters. This can occur for other "pure extrapolation" settings where a low validation error does not always correspond to a low test error. In this case, if a small amount of data from the test space (which, in our case, were C2 and C3 species) can be obtained, that can be included in the validation set to tune the hyperparameter more effectively. The problem setting, however, would no longer be a pure extrapolation as a small amount of data points from the test space is included during the training phase.

Figure 7 shows the values of the learned weights between the input layer and the hidden layer for each filter when eight filters were used. For each matrix, a row corresponds to the 10 weights going out of one fingerprint value (see Figure 4). A column corresponds to the eight values going into a hidden layer unit. There are three key observations here. First, each filer learns a different function of its inputs. Second, the sixth and seventh rows contain weights with high absolute values. This is because the weights were shared across the atom types, i.e, fingerprints from carbon and oxygen atoms were fed to the same filters, and according to Figure 4, some of the entries at the same location of the fingerprint vector for carbon and oxygen carry a different meaning. According to this, the fifth, sixth, and seventh entries have to encode more information, and hence the network learned higher valued weights for some of those. Finally, the fourth row for each of the filters learned close-to-zero weights. The fourth entry in the atomic fingerprint is the count of the number of carbon atoms to which the current atom is bonded where the carbon atom has three free valencies. In our training set, there was no such species. So, the network did not learn any significant value for those weights. This also signifies that if any species in the target set contains such a structure, its prediction would be inaccurate. Indeed, we found that two species, CH_2C and CH_3C (not included in the 29 species used as our target set),

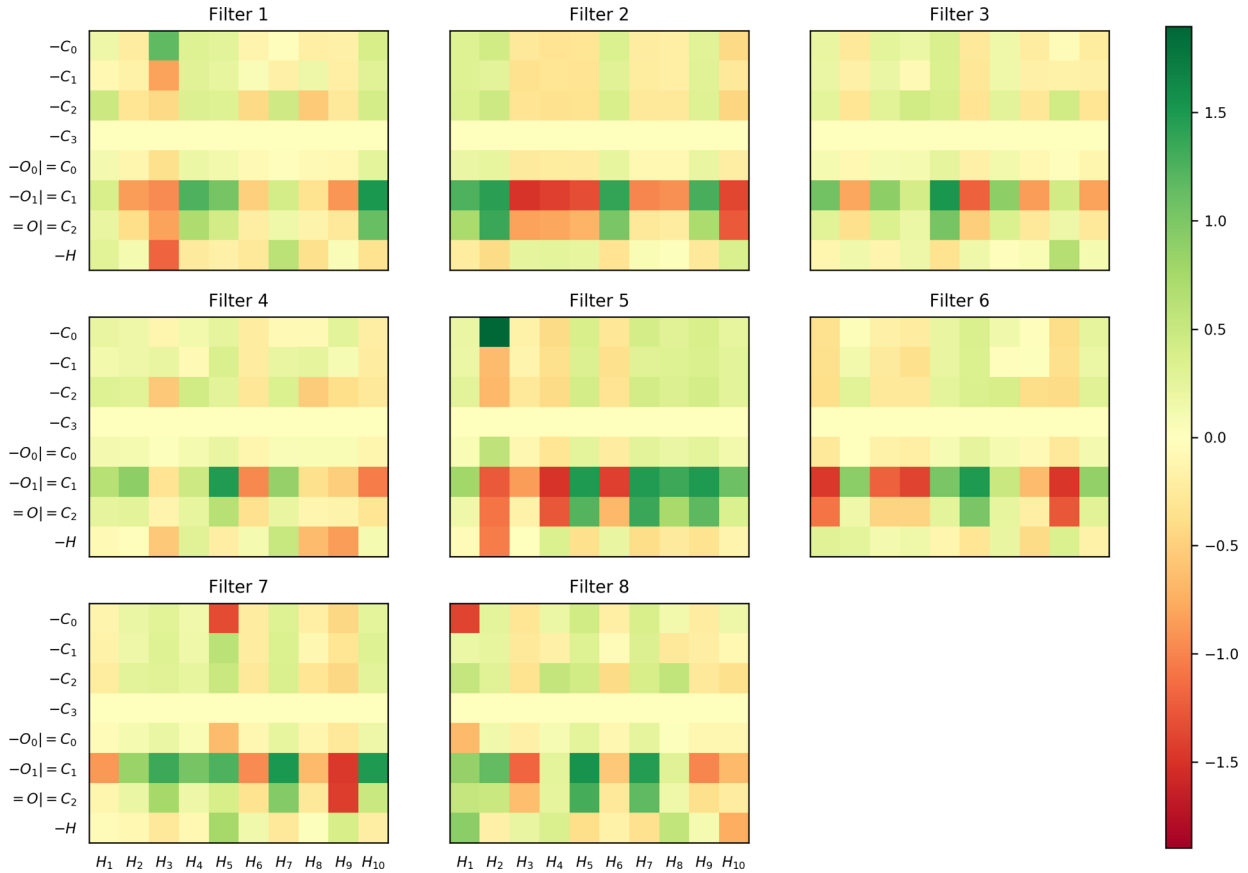


Figure 7. Weight matrices for the eight filters learned by running our proposed model. Each of the eight 8-by-10 matrices contains the values of the weights that connect each of the 8 atomic fingerprint values (which according to Figure 4 are the number of different bond types to which an atom is connected) to the 10 hidden units denoted as H_1−H_{10} (the left half of each subnetwork or filter shown in Figure 5).

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114

both containing this type of structure, had very high prediction errors (around 1 eV). This observation signifies a fundamental limitation in machine learning based models—the predictions can be at most as good as the data that are fed to train the model. The model learns from the training data the adsorption energy as a function of the basic building blocks or fragments that make up a chemical species, such as the information on how many free valencies an atom has or how many unsaturated carbon atoms an atom is connected to, etc. The effectiveness of the model, hence, is dependent on how well the encoded structural information in the fingerprint can describe the differences between the target property (here, adsorption energy) among different chemical species. Fragments that are absent in the training data, such as aromatic rings, are not learnt by the model, and thus the model will likely fail for species containing such fragments.

4. CONCLUSION

In this paper, we have performed a detail investigation on a predictive model for both interpolation and extrapolation of adsorption energies of hydrocarbon species on Pt(111) catalyst surface. We have compared the effectiveness of different fingerprints and ML models. For interpolation, our results indicate that a simple SMILE based fingerprint calculated from nearest neighbors with kernel based ML models perform very well for interpolation of adsorption energies with an MAE of 0.14 eV. However, when predicting adsorption energies of species of different size from that of the training set (extrapolation), only an additive atomic contribution based model works reasonably well. To improve upon this method, we have developed a multifilter based weighted additive model that combines the established additive model with the concept of filters from a convolutional neural network. Our findings show that this approach is highly generalizable compared to other models and leads for extrapolation of adsorption energies to an MAE of 0.23 eV. The proposed model also worked well when applied to interpolation with no statistically significant difference with the best models. The model has the potential to be applicable in other problems if the hyper-parameters of the model are adjusted according to the task. In the current work, all species were chain structures and of size between C1 and C4. The model was able to successfully extrapolate from larger to smaller species as long as the target species had similar chemical fragments as those in the training set. However, it should be noted that for distant extrapolation from small species to large species such as enzymes and proteins, the relative number of atoms bonded to the surface might be different, and the model needs further refinement for such scenarios.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.9b00986.

Data for energy and SMILES for species; calculation method and descriptor values for CM, BoB, flat molecular fingerprints, and atomic fingerprints; hyper-parameters for the proposed model; tables for complete results of interpolation and extrapolation (ZIP)

■ AUTHOR INFORMATION

Corresponding Authors

Andreas Heyden − *Department of Chemical Engineering, University of South Carolina, Columbia, South Carolina 29208, United States;* orcid.org/0000-0002-4939-7489; Email: heyden@cec.sc.edu

Gabriel A. Terejanu − *Department of Computer Science, University of North Carolina at Charlotte, Charlotte, North Carolina 28223, United States*; Email: gterejan@uncc.edu

Authors

Asif J. Chowdhury − *Department of Computer Science and Engineering, University of South Carolina, Columbia, South Carolina 29208, United States*

Wenqiang Yang − *Department of Chemical Engineering, University of South Carolina, Columbia, South Carolina 29208, United States*

Kareem E. Abdelfatah − *Department of Computer Science and Engineering, University of South Carolina, Columbia, South Carolina 29208, United States*

Mehdi Zare − *Department of Chemical Engineering, University of South Carolina, Columbia, South Carolina 29208, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.9b00986

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work has been supported by the National Science Foundation under Grant No. DMREF-1534260 (the bulk of the machine learning research and the DFT data on the HDO of succinic acid) and the U.S. Department of Energy Office of Science, Office of Basic Energy Sciences, Catalysis Science Program under Award DE-SC0007167 (most of the DFT data on the HDO of propanoic acid). K.E.A. acknowledges financial support from the National Science Foundation under Grant No. OIA-1632824. Computational resources provided by XSEDE facilities located at San Diego Supercomputer Center (SDSC) and Texas Advanced Computing Center (TACC) under grant number TG-CTS090100, U.S. Department of Energy facilities located at the National Energy Research Scientific Computing Center (NERSC) under Contract No. DE-AC02-05CH11231, Pacific Northwest National Laboratory (Ringgold ID 130367, Grant Proposal 49246), and the High-Performance Computing clusters located at University of South Carolina are gratefully acknowledged.

■ REFERENCES

(1) Nørskov, J. K.; Studt, F.; Abild-Pedersen, F.; Bligaard, T. *Fundamental Concepts in Heterogeneous Catalysis*; John Wiley and Sons: Hoboken, NJ, 2014; Chapter 2, pp 17−19.

(2) Bo, C.; Maseras, F.; López, N. The role of computational results databases in accelerating the discovery of catalysts. *Nat. Catal.* **2018**, *1*, 809−810.

(3) Nørskov, J. K.; Abild-Pedersen, F.; Studt, F.; Bligaard, T. Density Functional Theory in Surface Chemistry and Catalysis. *Proc. Natl. Acad. Sci. U. S. A.* **2011**, *108*, 937−943.

(4) Busch, M.; Wodrich, M. D.; Corminboeuf, C. Linear scaling relationships and volcano plots in homogeneous catalysis - revisiting the Suzuki reaction. *Chem. Sci.* **2015**, *6*, 6754−6761.

(5) Abild-Pedersen, F.; Greeley, J.; Studt, F.; Rossmeisl, J.; Munter, T. R.; Moses, P. G.; Skúlason, E.; Bligaard, T.; Nørskov, J. K. Scaling

properties of adsorption energies for hydrogen-containing molecules on transition-metal surfaces. *Phys. Rev. Lett.* **2007**, *99*, 016105.
(6) Greeley, J. P. Theoretical heterogeneous catalysis: scaling relationships and computational catalyst design. *Annu. Rev. Chem. Biomol. Eng.* **2016**, *7*, 605−35.
(7) Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; Vaughan, J. W. A theory of learning from different domains. *Mach. Learn.* **2010**, *79*, 151−175.
(8) Lu, J.; Behtash, S.; Faheem, M.; Heyden, A. Microkinetic modeling of the decarboxylation and decarbonylation of propanoic acid over Pd(111) model surfaces based on parameters obtained from first principles. *J. Catal.* **2013**, *305*, 56−66.
(9) Behler, J.; Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **2007**, *98*, 146401.
(10) Pereira, F.; Xiao, K.; Latino, D. A. R. S.; Wu, C.; Zhang, Q.; Aires-de Sousa, J. Machine learning methods to predict density functional theory B3LYP energies of HOMO and LUMO orbitals. *J. Chem. Inf. Model.* **2017**, *57*, 11−21 PMID: 28033004.
(11) Chowdhury, A. J.; Yang, W.; Walker, E.; Mamun, O.; Heyden, A.; Terejanu, G. A. Prediction of adsorption energies for chemical species on metal catalyst surfaces using machine learning. *J. Phys. Chem. C* **2018**, *122*, 28142−28150.
(12) Rupp, M.; Tkatchenko, A.; Müller, K.-R.; von Lilienfeld, O. A. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.* **2012**, *108*, 058301.
(13) Hansen, K.; Biegler, F.; Ramakrishnan, R.; Pronobis, W.; von Lilienfeld, O. A.; Müller, K.-R.; Tkatchenko, A. Machine learning predictions of molecular properties: accurate many-body potentials and nonlocality in chemical space. *J. Phys. Chem. Lett.* **2015**, *6*, 2326−2331.
(14) Behler, J. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *J. Chem. Phys.* **2011**, *134*, 074106.
(15) Morawietz, T.; Behler, J. A density-functional theory-based neural network potential for water clusters including van der Waals corrections. *J. Phys. Chem. A* **2013**, *117*, 7356−7366 PMID: 23557541.
(16) Behler, J. Perspective: machine learning potentials for atomistic simulations. *J. Chem. Phys.* **2016**, *145*, 170901.
(17) Ulissi, Z. W.; Tang, M. T.; Xiao, J.; Liu, X.; Torelli, D. A.; Karamad, M.; Cummins, K.; Hahn, C.; Lewis, N. S.; Jaramillo, T. F.; Chan, K.; Nørskov, J. K. Machine-learning methods enable exhaustive searches for active bimetallic facets and reveal active site motifs for CO2 reduction. *ACS Catal.* **2017**, *7*, 6600−6608.
(18) Rogers, D.; Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742−754 PMID: 20426451.
(19) Lo, Y. C.; Rensi, S. E.; Torng, W.; Altman, R. B. Machine learning in chemoinformatics and drug discovery. *Drug Discovery Today* **2018**, *23*, 1538−1546.
(20) Sanchez-Lengeling, B.; Aspuru-Guzik, A. Inverse molecular design using machine learning: generative models for matter engineering. *Science* **2018**, *361*, 360−365.
(21) Wu, Z.; Ramsundar, B.; Feinberg, E.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **2018**, *9*, 513−530.
(22) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *J. Comput.-Aided Mol. Des.* **2016**, *30*, 595−608.
(23) Duvenaud, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gómez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. *Proceedings of the 28th International Conference on Neural Information Processing Systems*, Cambridge, MA, 2015; Vol. 2, pp 2224−2232.
(24) Rupp, M.; Ramakrishnan, R.; von Lilienfeld, O. A. Machine learning for quantum mechanical properties of atoms in molecules. *J. Phys. Chem. Lett.* **2015**, *6*, 3309−3313.
(25) Segler, M. H. S.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **2018**, *4*, 120−131.
(26) Torng, W.; Altman, R. B. 3D deep convolutional neural networks for amino acid environment similarity analysis. *BMC Bioinf.* **2017**, *18*, 302.
(27) Collins, C. R.; Gordon, G. J.; von Lilienfeld, O. A.; Yaron, D. J. Constant size descriptors for accurate machine learning models of molecular properties. *J. Chem. Phys.* **2018**, *148*, 241718.
(28) LeCun, Y.; Haffner, P.; Bottou, L.; Bengio, Y. Object recognition with gradient-based learning. *Shape, Contour and Grouping in Computer Vision*: Springer Verlag: London, UK, 1999; pp 319−345.
(29) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84−90.
(30) Hubel, D. H.; Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *J. Physiol.* **1959**, *148*, 574−591.
(31) Ringach, D. L. Mapping receptive fields in primary visual cortex. *J. Physiol.* **2004**, *558*, 717−728.
(32) Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278−2324.
(33) Choromanska, A.; Henaff, M.; Mathieu, M.; Arous, G. B.; LeCun, Y. The loss surfaces of multilayer networks. *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, San Diego, CA, 2015; pp 192−204.
(34) Sutskever, I.; Martens, J.; Dahl, G.; Hinton, G. On the importance of initialization and momentum in deep learning. *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, GA, 2013; pp 1139−1147.
(35) Hanin, B.; Rolnick, D. In *Advances in Neural Information Processing Systems 31*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, 2018; pp 571−581.
(36) Vehbi Olgac, A.; Karlik, B. Performance analysis of various activation functions in generalized MLP architectures of neural networks. *Int. J. Artificial Intell. Expert Syst.* **2011**, *1*, 111−122.
(37) Tan, T. G.; Teo, J.; Anthony, P. A comparative investigation of non-linear activation functions in neural controllers for search-based game AI engineering. *Artif. Intell. Rev.* **2014**, *41*, 1−25.
(38) Maas, A. L.; Hannun, A. Y.; Ng, A. Y. Rectifier nonlinearities improve neural network acoustic models. *Proc. ICML* **2013**, *30*, 3.

https://dx.doi.org/10.1021/acs.jctc.9b00986
J. Chem. Theory Comput. 2020, 16, 1105−1114