Computational Materials Science 184 (2020) 109876

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Decision tree driven construction of rate constant models: Identifying the "top-N" environment atoms that influence surface diffusion barriers in Ag, Cu, Ni, Pd and Pt

Sandip Sawarkar, Abhijit Chatterjee*

Department of Chemical Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India

ARTICLE INFO

Keywords:
Decision trees
Cluster expansion models
Neural networks
Activated process
Molecular simulations
Machine learning

ABSTRACT

The local chemical environment is known to influence the rate constants of thermally activated atomic-scale processes in materials. In situations where rate constants vary over several orders of magnitude, dynamical materials simulations often require accurate rate constant models that can rapidly predict the rate for vast number of environments. Deriving rate constant models by fitting to a database of barriers can be particularly challenging when several environment atoms are believed to affect the rate. Previously, artificial neural networks (ANN) and cluster expansion models (CEM) have been employed as rate constant models. We demonstrate that a decision tree (DT) can complement training of such models by providing useful inputs. DTs can be used to (i) determine the relevant chemical environment, (ii) estimate the accuracy expected from CEM/ANN, (iii) identify cluster sizes required in CEM or size of the input layer in ANN so that the CEM/ANN model can be trained in a single step, and (iv) determine the minimum amount of data required for accurate training. Using this strategy, we construct for the first time CEM and ANN models for the exchange move (surface diffusion of metal on metal) that are both compact and accurate. The use of DT has enabled inclusion of large clusters, as big as 11 atom clusters in the CEM. Our strategy paves way for coupling DT and CEM/ANN for building computationally inexpensive rate constant models.

1. Introduction

In many real-world applications, understanding how the solid-state materials may structurally evolve over long periods of time is crucial towards engineering the performance, durability and safety of devices. In recent years, there has been surge of interest in accurately modeling changes happening to the material structure at the atomistic scales. One aspect involves attempting to build models that parametrize rate constants of atomistic processes in terms of the chemical environment. We refer to these types of models as rate constant models.

While several rare-event simulation techniques [1–5] can accurately provide rates, they are computationally too expensive for frequent calculation of rates with many chemical environments. In such situations, a rate constant model can step in. Rate constant models often lie at the heart of dynamical simulation techniques, such as kinetic Monte Carlo [6–9]. Unfortunately, in most materials the underlying interactions are complicated, and the functional form of the rate constant model is not available. For this reason, high-fidelity rate constant models have been largely constructed in terms of cluster expansion models (CEM) [10], artificial neural networks (ANN) [11], and other approaches [12,13] that offer flexible functional forms with a large number of fitting parameters that can be trained to multidimensional barrier data. In this work, we explore the use of decision tree (DT) [14] for the problem of rate constant model construction. We demonstrate that combining DTs with CEM/ANN can greatly simplify the overall model construction procedure.

Diffusion is an important atomistic scale mechanisms in solid state materials [15]. Here, we study surface diffusion in Ag, Cu, Ni, Pd and Pt. These metals along with their bimetallic alloys are used as catalysts. Surface diffusion processes need to be considered in areas of catalysis, corrosion and thin film growth [16–20]. Accelerated molecular dynamics (MD) simulations have revealed the presence of a variety of thermally activated processes at metal surfaces, such as hop, exchange and more complicated many-atom events, that contribute to surface diffusion [21,22]. Although accurate rate constant models for hop moves are available in literature, such is not the case for exchange moves. The temperature-dependent rate constants obey Arrhenius law, namely,

* Corresponding author.
E-mail address: abhijit@che.iitb.ac.in (A. Chatterjee).

https://doi.org/10.1016/j.commatsci.2020.109876
Received 9 April 2020; Received in revised form 10 June 2020; Accepted 10 June 2020
Available online 24 June 2020

$$k(\epsilon) = \nu(\epsilon)\exp\left(-\frac{E_a(\epsilon)}{k_B T}\right). \quad (1)$$

Here ν is the pre-exponential factor and E_a is the activation barrier, T denotes the absolute temperature, and k_B is the Boltzmann constant. Both ν and E_a depend on the local environment ϵ. Equation (1) is a rate constant model, however, the effect of the chemical environment ϵ on ν and E_a is not known. To simplify the problem, it helps to realize that ν may typically vary in the range of 10^{12} –10^{14} s^{-1}, while E_a may vary between ~ 0–2 eV [23]. Using Equation (1) we find that $k(\epsilon)$ can vary over several orders of magnitude. Considering a chemical uncertainty of $0.1eV$ in interaction models, ν is commonly assumed to be independent of the environment. Following this practice, we choose to focus only on $E_a(\epsilon)$.

The main issues while constructing a rate constant model are:

(a) How large is the chemical environment ϵ, and identifying which combinations of environment atoms are most relevant towards the calculation of E_a?
(b) What form of the model should be used? For e.g., should one truncate the CEM or how large should be the input layer in ANN (see later discussion)?
(c) How to balance the model complexity with model cost and accuracy.
(d) How many data points are necessary for training the rate constant model?

Here model complexity implies the number of environment atoms included in the model. A model made complex by including more environment atoms without significantly improving to its accuracy adds to the cost of using the model. Model cost also depends on the functional form of the model. Two models can have the same complexity, yet the computational cost may differ. Since typically CEM and ANN models are fitted to activation barrier databases constructed with a large number of environments, the model accuracy is measured in terms of goodness of fit. The objective is to generate accurate rate constant models with minimum complexity and cost.

In previous approaches, all issues (a)-(d) were handled simultaneously while training CEM or ANN, making the procedure inherently complicated. Here we divide the problem into two separate parts. Issue (a) is handled using DTs. Construction of a DT involves single iteration of training, testing and validation. We construct activation barrier databases for the exchange move using the nudged elastic band (NEB) method [5]. A DT involves a binary tree structure where tests involving the environment atoms are performed at each node. Tests involving the most important environment atoms are performed at the top-most nodes of the tree. Thus, the important features required in the rate constant model are readily available once the DT has been trained. Since the principle of the DT involves information gain, bottom levels of the tree explain smaller variability in the data [14]. Such levels may be excluded from CEM/ANN training. DT can rank environment atoms according to their importance. Even before the CEM/ANN has been trained, some idea about the loss of accuracy due to removal of the less important environment atoms can be gained from the analysis of the DT. Some aspects involving issue (b)-(d) can also be handled using DTs as discussed later. The second part of the problem involves issue (c), which addressed by training a CEM and ANN model using the top-most important environment atoms. The resulting CEM and ANN models of a fixed complexity are compared for their accuracy and cost.

The DT approach presented here can complement existing CEM and ANN construction procedures. Therefore, a brief review of previous rate model construction approaches, and the main issues therein, is provided in Section 2. Next, we outline our methodology in Section 3. Results and discussion are provided in Section 4. Finally, conclusions are presented in Section 5.

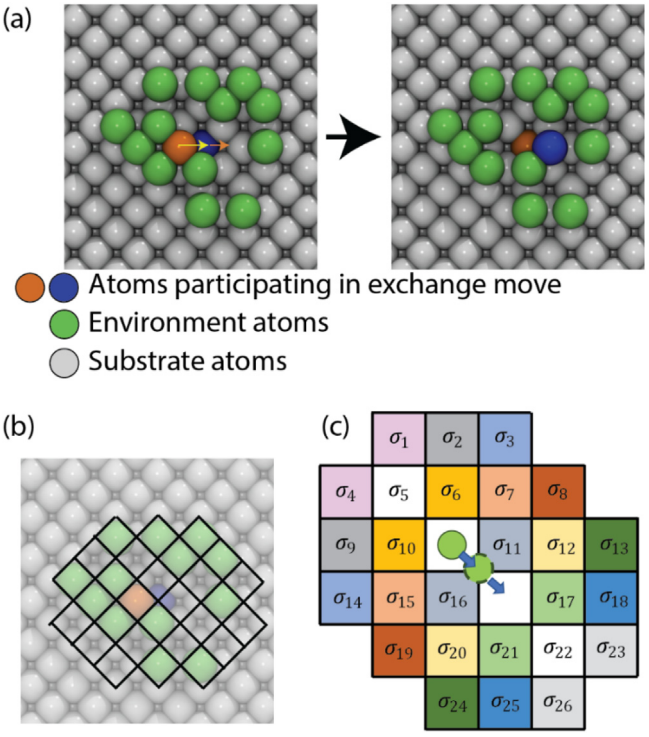

Fig. 1. (a) Example of an exchange move. Left and right panel are initial and final states. (b) Initial size of the environment used in the training of ANN. Initial state from (a) is shown for reference. (c) Lattice site numbering used in this work for creating barrier database, and subsequently for training DT and ANN model. Color denotes symmetrically equivalent sites.

2. Issues with standard approaches for construction of rate constant models

Fig. 1a shows an example of the exchange move. The orange (adsorbed) atom and blue (surface) atom participate in the move. We consider the chemical interaction effect of the environment atoms in the adlayer (green color) on surface diffusion via exchange mechanism. In crystalline materials, atoms reside on well-defined lattice sites. The occupation at a lattice site j is $\sigma_j = 1$ when it is occupied, and 0 otherwise. There are N_ϵ sites in the environment ϵ. Fig. 1b shows an initial guess for the size of environment based on the interaction cutoff. Note that all sites may not be relevant to the problem. Here, $N_\epsilon = 26$. Each square denotes a site. Fig. 1c shows the occupation variable at each site. As an example, in Fig. 1a $\sigma_1 = 1$ and $\sigma_4 = 0$. Adatoms outside the environment are assumed to be unimportant. The central question as far as the model complexity is concerned is how do we determine which of these N_ϵ sites are relevant to the exchange move?

2.1. Cluster expansion model

CEMs have been used extensively with crystalline materials to describe any property of interest Ξ as a function of the occupation vector $\sigma = (\sigma_1, \sigma_2, \cdots, \sigma_{N_\epsilon})$ [24,25]. A CEM is an expansion containing clusters of environment atoms: singlet, pairs, triplets and so on, i.e.,

$$\Xi(\sigma) = \xi_0 + \sum_i \xi_i\sigma_i + \sum_{\langle i,j\rangle} \xi_{ij}\sigma_i\sigma_j + \sum_{\langle i,j,k\rangle} \xi_{ijk}\sigma_i\sigma_j\sigma_k + \quad (2)$$

Here $\langle i, j, k\rangle$ implies that the triplet appears only once in the expansion. Sites in a cluster usually lie in vicinity of each other. The cluster terms are regarded as basis functions. For lattice systems, a CEM containing all possible clusters is exact. However, CEMs are usually truncated and larger clusters are excluded because of the high computational cost of training models with large-sized clusters.

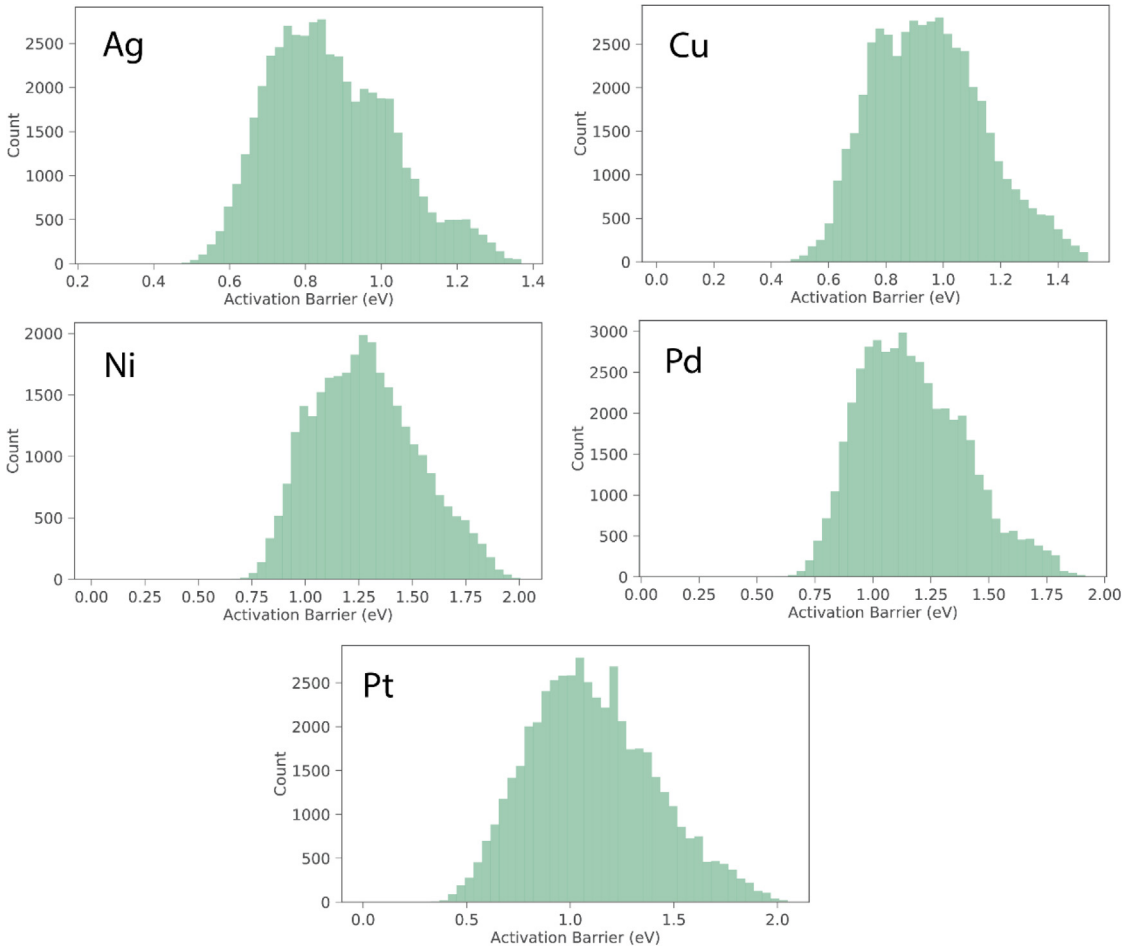

Fig. 2. Distribution of activation barriers collected using nudged elastic band calculations.

Cluster interactions ξ (CI) are determined using regression when a database containing E_a for several different ϵ is available. Generally, CEMs containing up to triplets, i.e.,

$$E_a(\epsilon) = e_0 + \sum_{i\in\epsilon} e_i\sigma_i + \sum_{\langle i,j\rangle\in\epsilon} e_{ij}\sigma_i\sigma_j + \sum_{\langle i,j,k\rangle\in\epsilon} e_{ijk}\sigma_i\sigma_j\sigma_k, \tag{3}$$

are considered. Symmetrically equivalent clusters possess identical values of the CI, e.g., for Fig. 1c the pair CIs $e_{6,11} = e_{10,16}$. This symmetry requirement arises from the underlying crystalline structure. Fig. 1c shows symmetric site occupations with the same color coding. Except for σ_5 and σ_{22} each occupation variable has a symmetric pair. CEM have been successfully applied to hop moves in pure metals [10,26]. These CEMs typically containing between 15 and 50 CIs. In comparison, the CEM is found to be relatively less accurate for exchange moves, even when triplets are included [27]. This may arise due to absence of large-sized clusters (size greater than 3) from the truncated CEM. Training a CEM with large clusters is computationally prohibitive. The number of CIs associated with clusters of size r grows combinatorially as $N_\epsilon!/r!(N_\epsilon - r)!$. Inclusion of large clusters comes at a price. First, optimizing CI parameters becomes difficult. Second, larger training databases are needed. In our experience, convergence to an optimal solution may become challenging once few thousands of CIs are present in the model. Reducing the model complexity by choosing the smallest possible ϵ, i.e., smallest N_ϵ, may help lower the optimization cost. However, systematic approaches to ascertain the smallest ϵ are unavailable. Consequently, iterative training of the CEM has been attempted wherein weak CIs (cluster interactions smaller than 0.01 eV in magnitude) are pruned over several iterations of model training. The final pruned CEM may contain less than 100 CIs even when thousands of CIs were present to begin with. For these reasons, cluster sizes beyond 3 are generally not attempted. As we shall show these issues are resolved using DTs.

During training the objective is to minimize the sum of squared errors

$$\min_p \sum_{\epsilon\in Tr} (E_a^{NEB}(\epsilon) - E_a^{model}(\epsilon; p))^2. \tag{4}$$

Here E_a^{NEB} and E_a^{model} denote the barrier calculated using NEB and the trained model, respectively, p are the model parameters and Tr denotes the training dataset.

2.2. Artificial neural network

For bulk materials N_ϵ greater than 100 has been previously considered with ANN [11,28]. In ANN, the site occupations are provided as an input layer to one or more neurons/activation functions. The structure of the ANN and the total number of weights is determined by the number of hidden layers and number of neurons per layer. One hidden layer is generally used for ensuring compact form of the ANN model. Ideas similar to ones discussed for CEM are used to estimate the ANN weights while training ANN models. As in CEM training, weights are optimized with the help of a database of $E_a(\epsilon)$. To overcome difficulties arising from large number of environment atoms, ideas such as constructing the ANN progressively in steps by adding environment atoms and/or neurons to the hidden layers have been explored [11]. In addition, one may include input variables into the ANN model one group at a time to identify groups of variables that improve the accuracy [11]. Rotational invariance of the environment can be introduced using spherical harmonics [28]. An alternate approach to impose

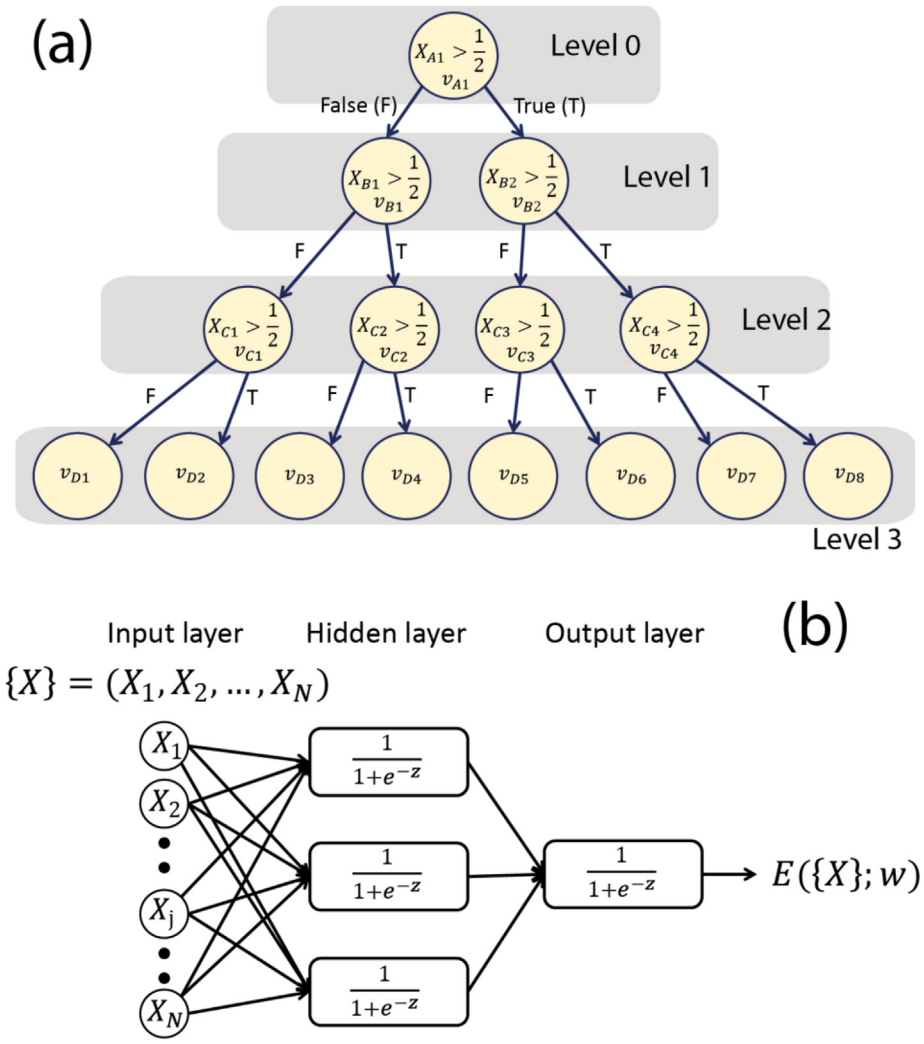

Fig. 3. Architecture of (a) DT and (b) artificial neural network used here. Sigmoid function is used as activation function in our ANN model. In panel (b), the input layer of ANN contains only N out of the 26 environment atoms and 3 neurons in the hidden layer.

symmetry constraints is introduced in Section 4 wherein a penalty function $\Lambda(p)$ is employed that ensures that symmetrically equivalent weights are equal.

2.3. Insights gained using decision trees

While large CEM/ANN models may provide better accuracy, a simple form of the rate constant models is of practical usefulness to dynamical methods such as KMC. The key development in this work is the application of DT to rank important sites, i.e., how sensitive the barriers are to lattice sites. This helps in achieving a compact rate constant model while maintaining high accuracy. Recall ANN models are often constructed in stages by including groups of sites. From chemical intuition we expect first nearest neighbor (1NN) sites to be more important than 2NN and far away sites. Therefore, a preliminary ANN model is often constructed by including only 1NN sites, assuming all 1NN sites to be equally important. Next, 2NN sites are included in the training procedure and so on, till the ANN model is found to be reasonably accurate. Unfortunately, in some situations our chemical intuition may be flawed. Later we shall demonstrate examples with the exchange move where 2NN sites are found to be more important than the 1NN sites defying our chemical intuition. Unimportant sites once included in the ANN remain in the model resulting in a higher complexity without any benefit to the model accuracy. ANN models have not been constructed for exchange moves, whereas CEMs have failed to provide desired accuracy. To our knowledge, a direct comparison between ANN and CEM has not been previously performed.

Another issue is to determine whether a database containing total M data points as sufficient for training, testing and validation. Suppose more data points are added so that the database now contains M' points. This may include new types of environment atoms. A newly trained CEM or ANN will contain new values of CIs or weights. Assessing whether the changes in CIs or weights are significant can be difficult. In comparison, the DT can be regarded as converged if the feature importance, average value of barrier and standard deviation with each test remains unchanged. These points are discussed later.

3. Methodology

3.1. Nudged elastic band calculations

A database of activation barriers for the exchange move is constructed for Ag, Cu, Ni, Pd and Pt (see Fig. 2). The Nudged Elastic Band (NEB) method [5] is used for the calculation of the activation barriers using the protocol discussed in Ref. [26]. In all NEB calculations, the same two atoms (adsorbed and surface atoms of Fig. 1a) participated in the exchange move. However, the adlayer occupations $\{\sigma\}$ in Fig. 1c varied from one NEB calculation to another. Out of the total number of 2^{26} environments possible, approximately 50,000 different $\{\sigma\}$ were randomly chosen for the construction of the database.

A $7 \times 7 \times 6$ unit cell slab was prepared. The slab contained two (1 0 0) surfaces along the z-direction. Bottom 4 layers for the slab were frozen. EAM potentials from Ref. [29] are used to describe the metal–metal interactions. Material properties like elastic constants, lattice

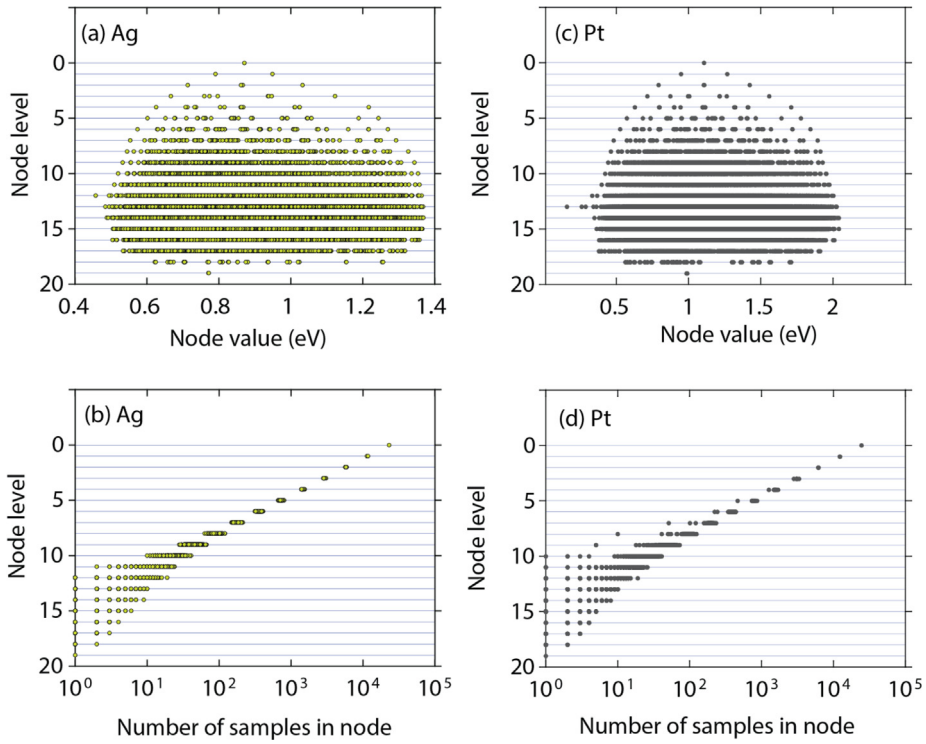

Fig. 4. (a) Average value of barrier, and (b) Number of samples at a node shown as a function of the levels in a DT fitted for Ag barrier database. (c) and (d) show corresponding values for Pt.

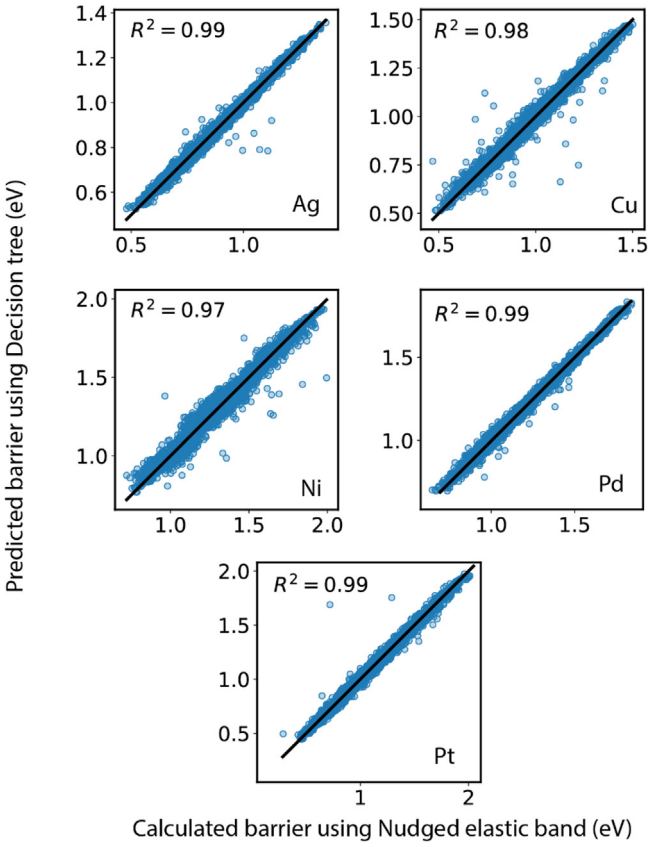

Fig. 5. Parity plot showing comparison of barriers from NEB and complete DT. A subset of the NEB data was used for training the DT. Plot shows comparison with validation dataset.

constants, bulk moduli, sublimation energies, phase behavior and vacancy formation energies are correctly obtained using the EAM parameterization [30,31]. The choice of the environment was dictated by the interaction cut-off. Typically, in EAM interactions are felt as far as 6 Å. Consequently, 1–4 nearest neighbor positions were included in the initial guess for the environment in Fig. 1c. NEB calculations were performed by first relaxing the initial and final states (energy minimization). Initially, 9 images were prepared by linearly interpolating positions of the atoms between initial and final states. Next, the minimum energy path was converged using a spring constant of 1 $eV/Å^2$. A global L-BFGS optimization scheme was employed. All calculations were performed using in-house codes. The saddle point energy from the climbing-image algorithm [5] was used to calculate the activation barrier. Data points where the number of transition states exceeded one were excluded. Environments resulting in activation barriers greater a maximum value were also removed from the database. The maximum barrier for Ag, Cu, Ni, Pd and Pt was 1.35, 1.5, 2, 1.9, 2.05 eV, respectively. The corresponding number of data points was 49267, 49600, 30129, 46581 and 49293. Fig. 2 shows the distribution of barriers for the different metals.

3.2. Decision tree

The information content at each site is a binary (0 or 1), indicating whether an atom is present or absent. Therefore, a decision tree is a suitable algorithm for classifying and regressing the effect of environment atoms. Fig. 3a shows a typical DT structure with multiple nodes and levels. Each node λ involves a test for a feature X_λ. Here feature implies one of the occupation variables σ_1, σ_2, …, or σ_{26}. Environment atoms resulting in the largest reduction in variance in data are present at the top of the tree. Exactly how an occupation variable is selected for a node will be discussed later. The top level (level 0) contains one node with the entire training dataset. At each decision node, the data is divided into smaller subsets based on the outcome of the test. For instance, in Fig. 3a samples that satisfy the condition $X_{A1} = 0$ end up in the node $B1$, while data in node $B2$ correspond to $X_{A1} = 1$. Similarly, at

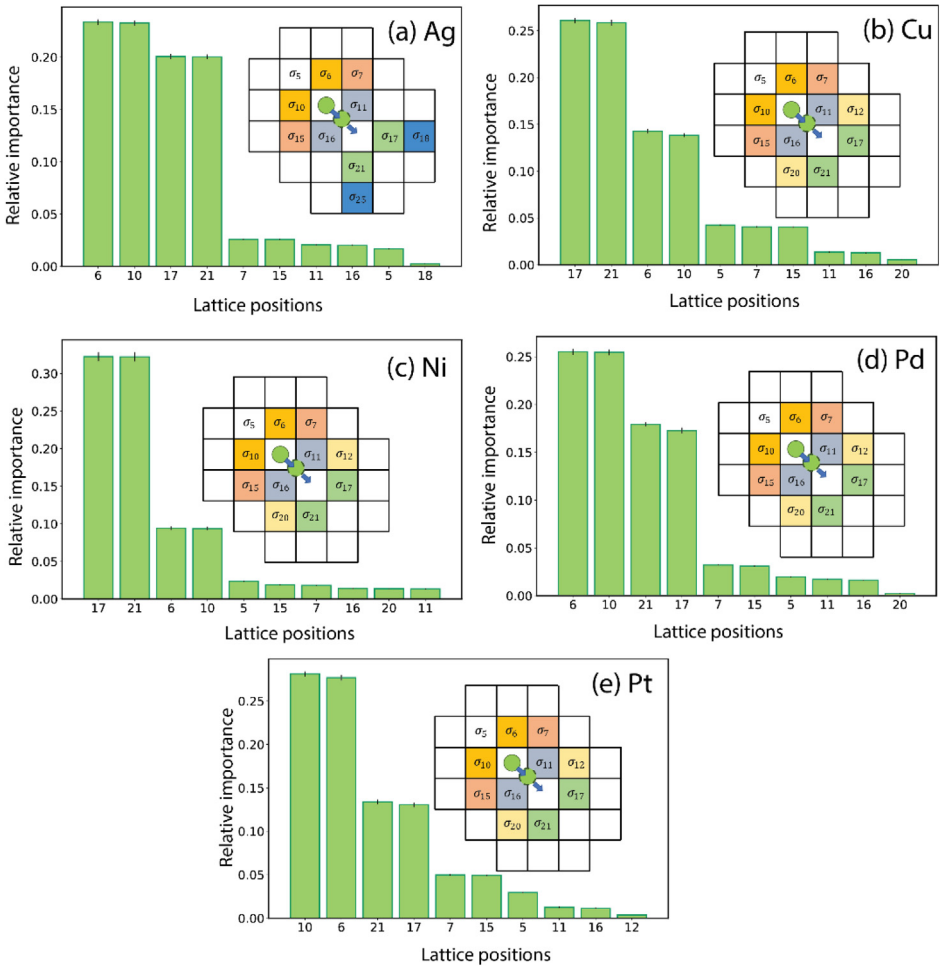

Fig. 6. Top-ten ranked features based on the relative feature importance in the trained DT when single sites are provided in list of features.

node $C4$ the data satisfies the condition $X_{A1} = X_{B2} = 1$. In general, each node λ contains a dataset d_λ with n_λ number of samples. The value v_λ for node λ is the average barrier calculated using d_λ, i.e.,

$$v_\lambda = \frac{1}{n_\lambda} \sum_{s \in d_\lambda} E_s. \tag{5}$$

Here E_s is the barrier for the sample $s \in d_\lambda$. The mean squared error (MSE) or equivalently the variance for the dataset is

$$MSE(\lambda) = \frac{1}{n_\lambda} \sum_{s \in d_\lambda} (E_s - v_\lambda)^2. \tag{6}$$

Suppose we are in the process of adding children to the node λ, and we need to decide which environment atom will correspond to feature X_λ. This decision is made based on the feature impurity for X at node λ, where X can be any occupation variable that possesses values 0/1 in the dataset d_λ. The putative impurity for a node is the weighted sum of MSE of children belonging to the node [14], i.e.,

$$I(\lambda; X) = \sum_{c \in \lambda} \frac{n_c}{n_\lambda} MSE(c) \tag{7}$$

where, c is a child node of λ. The occupation variable with the lowest impurity is selected as X_λ. Eventually, we write the node impurity $I(\lambda; X_\lambda) \equiv I(\lambda)$. The reduction in variance of children implies that the feature causing the large variation in d_λ has been eliminated, which has resulted in more homogenous sub-nodes. As a result, MSE typically decreases at lower down the DT. This procedure is recursed till a user defined stopping criterion is met. In our case, a node no longer possesses children once its sample size is 1.

When a feature X appears only once in the DT, e.g., $X_\lambda = X$, its feature importance (FI) is calculated as

$$FI(X) = n_\lambda (MSE(\lambda) - I(\lambda)). \tag{8}$$

The term in the bracket indicates the extent to the variance is reduced between the node λ and its children. More generally, a feature may appear multiple times in the DT. In such cases, the FI is calculated as the total contribution in reducing the variance

$$FI(X) = \sum_\lambda n_\lambda (MSE(\lambda) - I(\lambda)). \tag{9}$$

Finally, the relative feature importance is calculated as

$$RFI(X) = \frac{FI(X)}{\sum_{X'} FI(X')}. \tag{10}$$

5000 points were kept separately from the original dataset for validation. These points were used to calculate the coefficient of determination R^2 value for the model. 80% of the remaining data points were randomly selected for training, while remaining were used for testing. Scikit-learn 0.22.2 was used to implement the decision tree model [32].

Fig. 4 shows the typical behavior at nodes of a DT. Results are shown for two trees, one which was fitted to the Ag barrier database and another to Pt database in Section 4. Average barrier for nodes are scattered around the value at the top node (Fig. 4a and 4c). Size of the dataset (number of samples) at each node is shown in Fig. 4b and 4d. The sample size at each subsequent level decreases by a factor of 2 since training data is randomized, as evident from Fig. 4b and 4d. Eventually, the sample size at bottom of the tree becomes 1. The node RMSE decreases as we move down the levels (not shown).

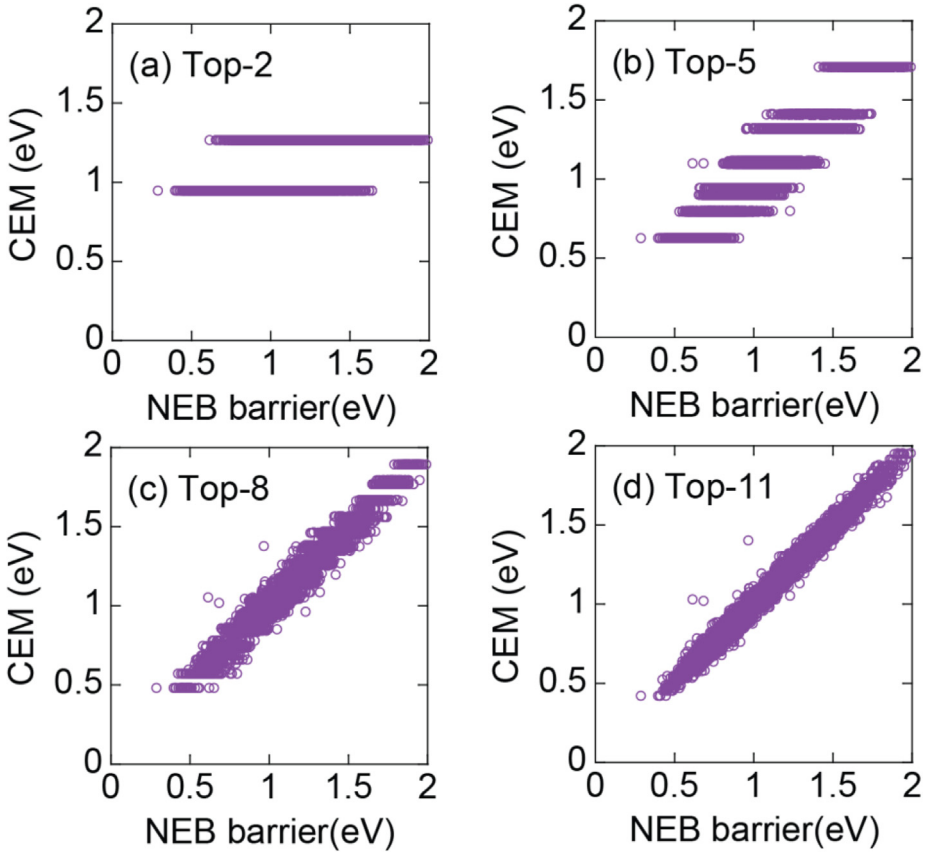

Fig. 7. Parity plots for Pt illustrating the effect on accuracy while adding top-N levels of the DT to construct the cluster expansion model. Plot includes training, testing and validation datasets.

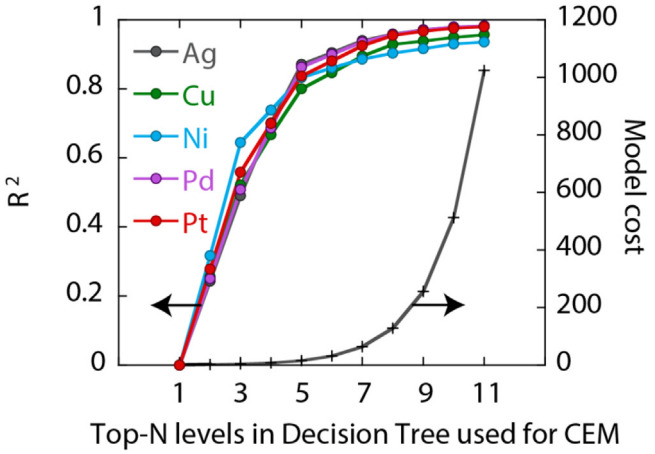

Fig. 8. Coefficient of determination (R^2) and model cost as a function of the top-N tree levels used to construct a cluster expansion model.

Let us we restrict ourselves to the top-three levels in Fig. 3a, i.e., levels 0–2 included. The DT can return one of four possible values for the barrier, i.e.,

$$E_a(X_{A1}, X_{B1}, X_{B2}) = v_{C1}(1 - X_{A1})(1 - X_{B1}) + v_{C2}(1 - X_{A1})X_{B1} \quad (11)$$

$$+ v_{C3}X_{A1}(1 - X_{B2}) + v_{C4}X_{A1}X_{B2}.$$

Equation (11) is equivalent to a cluster expansion model containing singlets and pair clusters, i.e.,

$$E_a(X_{A1}, X_{B1}, X_{B2}) = v_{C1} + (v_{C3} - v_{C1})X_{A1} + (v_{C2} - v_{C1})X_{B1} + \quad (12)$$

$$(v_{C1} - v_{C2})X_{A1}X_{B1} + (v_{C4} - v_{C3})X_{A1}X_{B2}$$

Suppose a CEM were to be created by truncating the DT till the top-N levels, the number of barrier values in the resulting CEM, e.g., v_{C1}, v_{C2}, $\cdots$ in Equation (11), will be 2^{N-1}. A CEM formed using the top-1, 2 and 11 levels will possess 1, 2 and 1024 values, respectively. The maximum number of (singlets and larger sized) clusters in the CEM is 2^{N-1} (see Equation (12)). Maximum cluster size in the CEM is $N - 1$, which can be much larger than cluster sizes traditionally handled with CEMs (see discussion around Equation (3)).

3.3. Artificial neural network

The ANN structure used in this work possesses one input layer, one hidden layer and one output layer. Each hidden layer contains N_h number of neurons. Fig. 3b shows an example with 3 neurons. The hidden layer transforms the input, while the output layer provides the model prediction. Assuming that the input layer possesses N input features, the weighted input from all features along with bias parameter is passed to each neuron in the hidden layer. The neuron activation function acts on linear combination of weighted input variable and bias, resulting in a nonlinear output from the neuron. We use the sigmoid function as an activation function. The output from the sigmoid function lies between 0 and 1. Accordingly, we normalize the barriers by dividing with the maximum barrier in the database before training the model. Finally, the output from the ANN is multiplied by the normalization factor to obtain the predicted barrier. Scikit-learn 0.22.2 was used for training the ANN model [32].

Initially top-N important features were identified using DT. The total number of ANN parameters including weights and bias are $N(N_h + 2) + 1$. The weights are estimated using the backpropagation method [33]. First, the database was randomly divided into three parts. As with DT training, 5000 data points were used for validation, 80% of

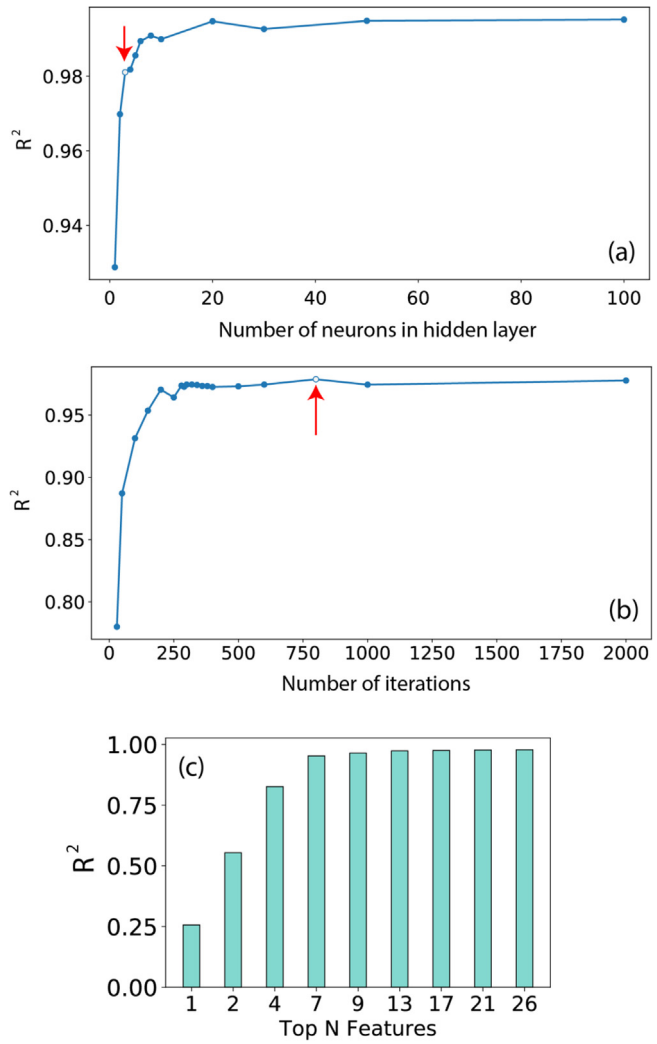

Fig. 9. Coefficient of determination R^2 for the ANN models constructed for Pt. Effect of (a) number of iterations and (b) number of neurons in the ANN hidden layer, and (c) top-N occupations from Fig. 6e included in the input layer are considered. Parameter value selected for remaining calculations is highlighted by the red arrow in each panel.

the remaining data was used for training and 20% for testing. More details on the choice of ANN training parameters are provided later.

4. Results and discussion

4.1. DT – Single sites as features

A total of 20 levels are present in the trained DT. Barriers predicted by the DT are in good agreement with the barriers calculated using NEB. Fig. 5 shows the parity plot of activation barriers. The coefficient of determination R^2 value obtained was between 0.97 and 0.99. In the CEM constructed by truncating the DT, interactions observed for symmetric positions in the local environment were nearly equal.

Fig. 6 shows the relative importance of lattice sites in the environment. The relative feature importance for symmetric sites are nearly equal. The most important symmetric pair is {6, 10} for Ag, Pd and Pt, whereas, {17, 21} for Cu and Ni. Other top sites (including symmetric pair) are also shown. Inset displays the positions of these sites around the exchanging atoms. Let us construct a distance measure to assess the proximity of a site to the exchanging atoms. The distance measure is based on sum of distances of the given site to the initial positions of the exchanging atoms. Using the distance measure, sites are arranged in order {11, 16} < {6, 10} < {7, 15} < {12, 20} < {17, 21} in terms of distance. Interestingly, the first nearest neighbor sites {11, 16} are relatively unimportant to the exchange move barrier. In case of Cu and Ni, the fifth nearest neighbor sites {17,21} are most important. These observations are contrary to our chemical intuition and provide an important lesson: one should use feature importance (FI) to determine sites that are statistically important and not distance measures. Despite these useful insights the large size of the complete DT makes it is unwieldy. For this reason, we construct a CEM by truncating the tree to the top-N levels.

4.2. Using top-N levels of DT to construct cluster expansion model

Fig. 7 shows the barriers predicted using CEMs for Pt constructed with top-N levels of the DT, $N = 2, 5, 8$ and 11, plotted against the computed barrier. For small N there are fewer barrier values that have to cater to a wide variety of environments. For instance, when $N = 2$ the CEM is given by $E_a(\sigma_{10}) = 0.948(1 - \sigma_{10}) + 1.267\sigma_{10}$. The model results in a large horizontal spread as seen in Fig. 7a, as only two values of barrier are possible. As N increases, more sites are included in the model, and as a consequence, the model becomes more complex. The CEM with top-11 levels contains all 26 sites. Now, diverse environments are handled in a better way, leading to almost continuum-like values of the barrier and better agreement with NEB.

CEMs constructed for Ag, Cu, Ni, Pd and Pt for $N = 1$ to 11 have been included as separate files (written in Fortran 90 language) in Section S1 of Supporting Information. The reader is instructed to briefly study these codes to appreciate the size of the models. The codes can be readily employed with KMC simulations as discussed in the Supporting Information. Fig. 8 provides an overview of the coefficient of determination (R^2) and model cost (in terms of number of clusters in the model). Similar behavior is observed for all metals. R^2, which is zero for $N = 1$, increases to 0.98 for Ag, Pd and Pt, 0.96 for Cu and 0.94 for Ni when $N = 11$. A comparison to Fig. 5 shows the marginal loss in accuracy by not considering all DT levels in the CEM. Since KMC simulations often access large length and timescales, computationally expensive CEMs will adversely affect the computational efficiency. Ultimately, the choice regarding the value of N is to be made by the user to strike a balance between accuracy, complexity and computational cost. Insights from relative feature importance (e.g., Fig. 6) can be taken into account while preparing a CEM. Such an attempt will be made while training the ANN model.

In our experience, including clusters directly as features in the DT does not work well. When large number of clusters, e.g., singlets, pairs and triplets, were included the total number of features increased to 2951. It is well known that decision trees tend to over-fit on data with a large number of features. When some clusters dominate the training data the resulting DT may end up becoming biased. Consequently, after training the DT and obtaining the relative feature importance, although some pair and triplet clusters were ranked as important, symmetrically equivalent clusters did not possess the same relative importance. See discussion in Section S2 of Supporting Information.

4.3. Using DT to ANN regression model

We assess accuracy and cost of ANN model constructed using top-N site occupations. To ensure that symmetric weights are equal we add a penalty function $\Lambda(p)$ where p are the weights in the ANN model. Equation (4) becomes

$$\min_p \left\{ \Lambda(p) + \sum\nolimits_{\epsilon \in Tr} \left(E_a^{NEB}(\epsilon) - E_a^{model}(\epsilon; p)\right)^2 \right\}. \tag{13}$$

The form of the penalty function is

$$\Lambda(p) = \sum\nolimits_{< p_x, p_y >} (p_x - p_y)^2 \tag{14}$$

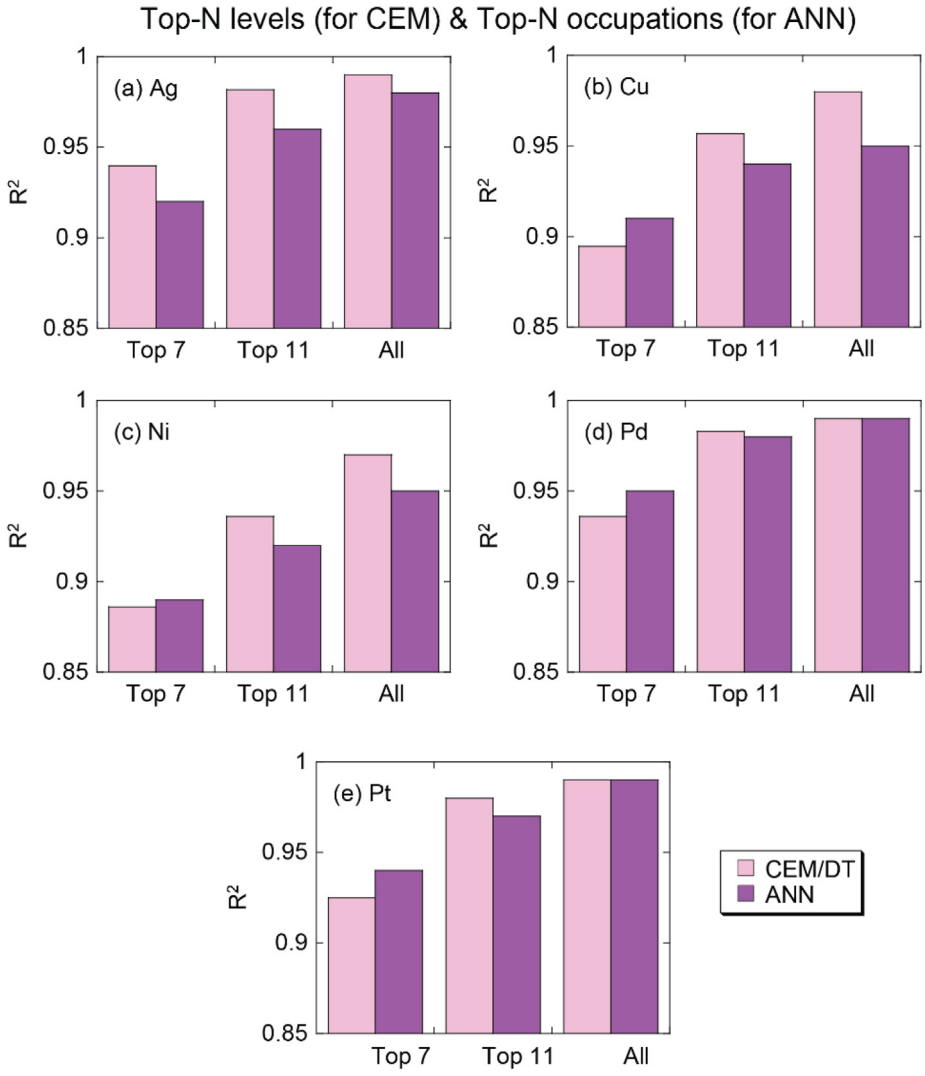

Fig. 10. Comparison of coefficient of determination R^2 value for CEM and ANN models.

Table 1
Weight and bias for hidden and output layer of ANN model using top-11 important sites. NF denotes normalization factor for the database.

		Hidden layer							Output layer	
		w_{X,N_i}						Bias b_{N_i}	Weight p_{N_i}	Bias q
Metal (NF)	**Neuron**	σ_5	σ_6/σ_{10}	σ_{17}/σ_{21}	σ_7/σ_{15}	σ_{11}/σ_{16}	σ_{12}/σ_{20}			
Ag (1.35 eV)	N_1	0.18	1.30	-1.28	0.26	-0.38	-0.09	-3.25	3.71	1.45
	N_2	-0.25	-1.03	-0.65	-0.43	-0.34	-0.31	1.48	-1.53	
	N_3	-0.13	-0.15	1.09	-0.04	0.37	0.16	-1.74	-1.73	
Cu (1.5)	N_1	0.23	0.2	-0.46	0.24	0.22	0.0	-0.55	3.62	0.06
	N_2	0.19	1.17	-1.54	0.39	-0.02	-0.19	-3.99	4.75	
	N_3	-0.12	-0.56	0.18	0.13	1.59	0.19	0.67	-1.49	
Ni (2.0)	N_1	0.11	0.22	-0.89	-0.01	-0.54	-0.27	-0.24	5.61	1.67
	N_2	-0.23	-0.65	-1.46	-0.36	-0.63	-0.46	0.73	-2.35	
	N_3	-0.14	-0.14	0.55	-0.28	-1.07	-0.06	2.89	-2.29	
Pd (1.9)	N_1	0.14	0.08	-0.99	0.03	-0.31	-0.16	1.33	2.05	-0.35
	N_2	0.17	1.32	-1.12	0.32	-0.28	0.07	-3.83	4.18	
	N_3	-0.17	-0.92	-0.63	-0.37	-0.24	-0.29	0.91	-2.14	
Pt (2.05)	N_1	0.27	1.21	0.74	0.58	0.59	0.47	-1.9	1.92	-0.67
	N_2	0.20	0.34	-0.89	0.15	-0.30	-0.20	-1.00	2.49	
	N_3	-0.38	-1.84	1.43	-0.52	0.23	0.13	4.58	-2.19	

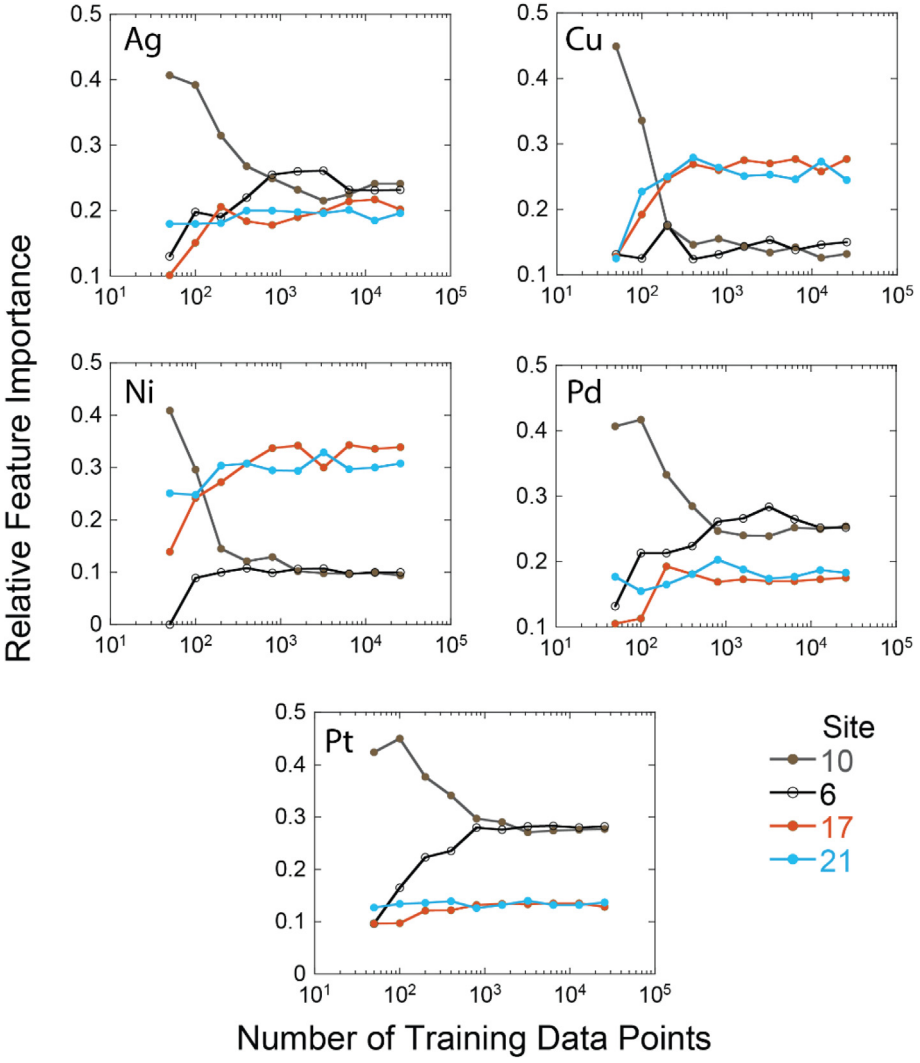

Fig. 11. Relative feature importance as a function of the number of samples in the training dataset.

and $<p_x, p_y>$ denotes a pair of symmetric weights.

The ANN training employs a number of parameters, namely, number of neurons in hidden layer, maximum number of iterations for optimization, and the size of the input layer, which need to be specified before we can proceed. The R^2 value for the simplest ANN model for Pt with input layer containing 26 occupations followed by one neuron in hidden layer and one output layer was found to be 0.93 (using 800 maximum iterations for optimization). Increasing the number of neurons in the hidden layer improves the fit as seen in Fig. 9a. $R^2 = 0.98$ or higher was deemed sufficient for regression. Based on this requirement, the number of neurons in the single hidden layer is selected as 3. Similarly, Fig. 9b shows the effect of maximum number of iterations for optimization. Earlier, from the analysis of the DTs we had ranked sites in order of their importance. Using this information, in Fig. 6f we determine the minimum number of sites that need to be included in the input layer to achieve $R^2 = 0.98$ or higher with Pt. ANN models for Pt were constructed with the top-N site occupations from Fig. 6f using different values of N. As expected, the R^2 value increases with N, as seen in Fig. 9c. For maintaining uniformity, the number of iterations, number of neurons and N were fixed at 800, 3 and 11, respectively, for the remainder of the study.

Fig. 10 shows for each metal a comparison of R^2 values for the fitted ANN and CEM of different complexities. The ANN model was constructed by choosing top-N occupation variables, whereas the CEM was constructed using the top-N levels. In most cases, the CEM appears to perform marginally better in terms of accuracy than the ANN when N is fixed. When $N = 11$, R^2 greater than 0.95 is achieved with Ag, Pd and Pt, whereas model accuracy with Cu and Ni is slightly less. Nonetheless, these models are significantly more accurate than previous models developed for exchange move [27]. One reason why CEM and ANN models exhibit similar trends for R^2 versus N is as follows. Recall that a DT truncated at level N can return one unique value of the barrier depending on which of the 2^{N-1} clusters (or equivalently nodes/branches as in Equation (11)) is present in a given occupation vector. The DT training process simply allows clusters to be chosen in a manner so that the reduction in variance is achieved in the best possible way. For an ANN constructed with top-N occupations, the input layer can possess 2^N different values. The ANN yields 2^N possible barrier values irrespective of how many neurons are present in the hidden layer. Increasing the number of neurons in the hidden layer simply allows the ANN to be more correlated to the training dataset.

The fitted ANN models are tabulated in Table 1. The normalization factor (in units of eV) used to scale barriers so that the scaled barriers lie between 0 and 1 is provided in the left-most column. Values of weights and biases for inputs σ_i to each of the (three) neurons in the hidden layer are provided in the middle columns (shaded gray). Weights for occupation variables that are not mentioned have a value of 0. Finally, the weights and bias for the output layer is mentioned in the right columns. Clearly, the ANN is significantly more compact than the corresponding CEM with $N = 11$ in Section 4.2. ANN models written in

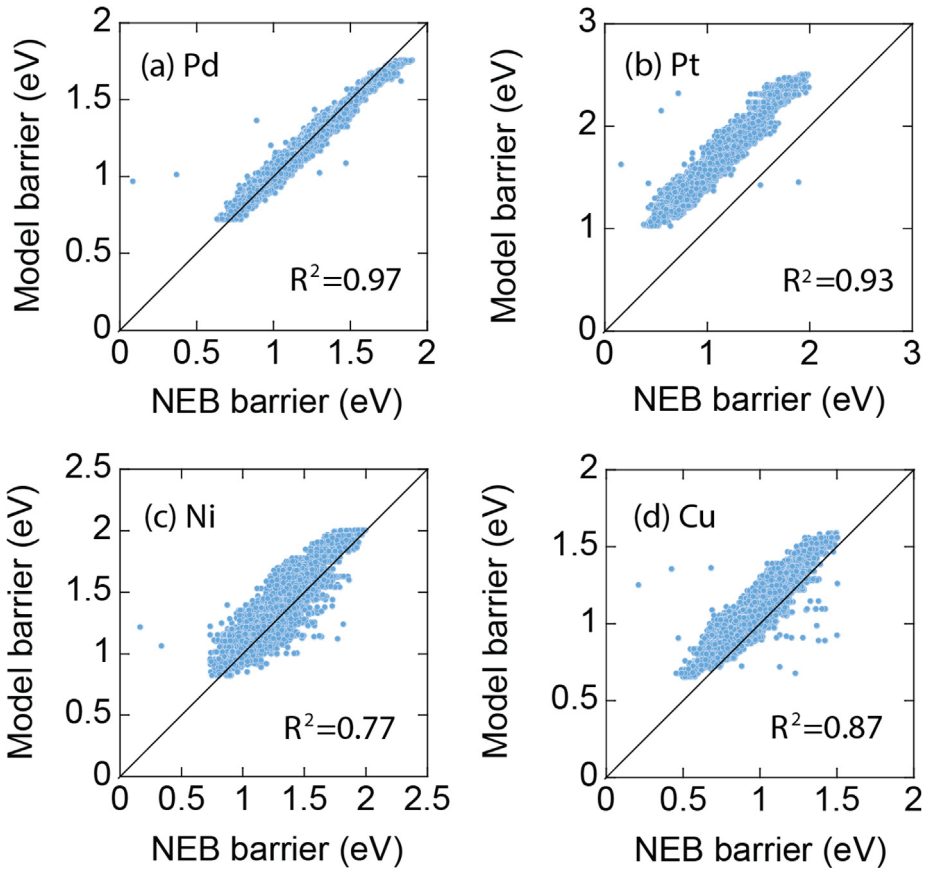

Fig. 12. Predictions for (a) Pd, (b) Pt, (c) Ni and (d) Cu barriers using ANN rate constant model constructed for Ag in Table 1. Predictions are made by scaling the Ag barrier by the cohesive energy of the metals involved.

Fortran language are provided in Section S3 of Supporting Information.

In summary, the steps involved in DT driven construction of CEM or ANN rate constant models are:

(i) Train a decision tree and ensure that the full DT is sufficiently accurate. The truncated DT or the ANN using a subset of the occupation variables will be less accurate.
(ii) Obtain relative feature importance based ranking of environment atoms.

To build a CEM we recommend that the top-N levels of the DT are utilized:

(i) Construct CEMs in form of truncated DT using different values of N, and obtain the coefficient of determination for each model.
(ii) Choose a value of N based on the cost, complexity and accuracy of the CEMs.

To build an ANN model we recommend that the top-N important sites in the DT are utilized:

(i) Construct ANNs using different values of N, and obtain the coefficient of determination for each model.
(ii) Choose a value of N based on the cost, complexity and accuracy of the ANNs.

We discuss the minimum amount of data required to train a DT. Variance in the overall data is largely independent of the training dataset size N_{Tr} when $50 \leq N_{Tr} \leq 25600$ (not shown). However, some features may dominate the dataset with small N_{Tr}. This can result in an inaccurate ranking of the features. Instead one should directly measure the reduction in the variance/MSE in terms of the relative feature importance. Recall in Fig. 4b and d, how the number of samples at each node decreases by nearly a factor of 2 as we step down to the lower levels of the DT. As a consequence, a DT with effectively $M = 5$ is obtained with $N_{Tr} = 50$. Suppose we require a DT with at least $0 - M$ levels, such that each node at the lowest level contains a minimum of four samples. With such a requirement, the number of samples in the training dataset should have been 2^{M+2}. The 2 in $M + 2$ arises because of the requirement of minimum four samples. For $M = 10$ (top-11 levels), which yielded a reasonably accurate CEM, the minimum training dataset size is $N_{Tr} = 2048$. Fig. 11 shows the RFI for the top four important sites, i.e., 6, 10, 17 and 21, as a function of N_{Tr}. When $N_{Tr} \leq 512$ a clear trend indicating changes in the RFI can be seen. Beyond $N_{Tr} \geq 2048$ the RFI converges to a value that is close to the one reported in Fig. 6. Some variations in RFI are expected due to the random selection of samples in the training set.

While applying machine learning techniques to material problems, it is a common practice to connect observed quantities to elemental properties of the material. Since the exchanging atoms participate in breaking of old bonds and formation of new ones, and the cohesive energy E_{coh} is a measure of the bond strength, one would expect some connection between $E_a(\epsilon)$ and E_{coh}. For the EAM potentials used in this paper, the cohesive energy of Ag, Cu, Pd, Ni and Pt is 2.85, 3.54, 3.91, 4.46 and 5.77 eV/atom, respectively. The average barrier calculated from Fig. 2 is 0.87, 0.95, 1.18, 1.28 and 1.11 eV for Ag, Cu, Pd, Ni and Pt, respectively. As expected, the average barrier is higher when E_{coh} is large. Suppose the barrier is proportional to the cohesive energy, then the normalized barrier obtained as $E_a(\epsilon)/E_{coh}$ for the different metals should be nearly equal. Using this rationale, we work with the Ag ANN model of Table 1 and scale the model output by the factor $E_{coh,metal}/E_{coh,Ag}$ to obtain a barrier prediction for a metal other than Ag.

Fig. 12 shows predicted barriers for Pd, Pt, Cu and Ni. The predictions obtained for Pd are well correlated with the NEB data. For Pt, although the R^2 value is 0.93 the barriers are overestimated. R^2 values for Ni and Cu are low. This observation is similar to the ones in Figs. 6, 11 and 12 where we find trends for Ag, Pt and Pd to be similar but different from Cu and Ni. In summary, differences in barriers in metals can be qualitatively traced to the E_{coh} values. However, rate constant model for one metal derived from a rate constant model of another metal simply by scaling the cohesive energies may not be accurate.

5. Conclusion

In this paper, we highlight decision trees (DTs) as a fast and effective tool for the construction of rate constant models based on cluster expansions, i.e., CEMs, and artificial neural networks (ANNs). The main advantage offered by DTs is that lattice sites within the environment are ranked in order of their importance. In absence of such information, CEM/ANN construction is iterative in nature and involves guessing the list of important environment atoms. DTs can be unwieldy because of their size. Consequently, they need to be coupled with CEM or ANN rate constant models. As shown in this paper, a truncated DT is equivalent to a CEM. Cluster sizes of up to 11 atoms were included in the model, which has improved the accuracy of the CEM. Similarly, analyzing the ANN model construction in terms of the top-N sites is useful as it can help strike a balance between accuracy, complexity and cost of the model. An ANN model trained using inputs from DTs is computationally less expensive than CEM with comparable accuracy.

In future, we hope to deploy our approach to diffusion processes in bimetallic alloys. We also expect the rate constant models to be coupled with KMC simulation of these metals. Diffusion in other material systems can also be studied with machine leaning techniques to bring down the overall computational cost.

6. Data availability

The raw data required to reproduce these findings are available to download from [INSERT PERMANENT WEB LINK(s)]. The processed data required to reproduce these findings are available to download from [INSERT PERMANENT WEB LINK(s)].

CRediT authorship contribution statement

Sandip Sawarkar: Writing - original draft, Data curation, Formal analysis, Methodology. **Abhijit Chatterjee:** Supervision, Conceptualization, Methodology, Writing - review & editing, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

AC acknowledges support from Science and Engineering Research Board, Department of Science and Technology Grant Nos. EMR/2017/001520 and MTR/2019/000909.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.commatsci.2020.109876.

References

[1] A.F. Voter, F. Montalenti, T.C. Germann, Extending the time scales in atomistic simulation of materials, Annu. Rev. Mater. Res. 32 (2002) 321–346.
[2] S. Divi, A. Chatterjee, Accelerating rare events while overcoming the low-barrier problem using a temperature program, J. Chem. Phys. 140 (2014) 184115, , https://doi.org/10.1063/1.4875476.
[3] R. Miron, K.A. Fichthorn, Accelerated molecular dynamics with the bond-boost method, J. Chem. Phys. 119 (2003) 6210–6216.
[4] G.T. Barkema, N. Mousseau, The activation–relaxation technique: an efficient algorithm for sampling energy landscapes, Comput. Mater. Sci. 20 (2001) 285–292, https://doi.org/10.1016/S0927-0256(00)00184-1.
[5] G. Henkelman, B.P. Uberuaga, H. Jónsson, A climbing image nudged elastic band method for finding saddle points and minimum energy paths, J. Chem. Phys. 113 (2000) 9901–9904.
[6] A. Kara, O. Trushin, H. Yildirim, T.S. Rahman, Off-lattice self-learning kinetic Monte Carlo: application to 2D cluster diffusion on the fcc(111) surface, J. Phys. Condens. Matter. 21 (2009) 84213.
[7] A. Chatterjee, D.G. Vlachos, An overview of spatial microscopic and accelerated Kinetic Monte Carlo methods, J. Comput. Mater. Des. 14 (2007) 253–308.
[8] O. Trushin, A. Karim, A. Kara, T.S. Rahman, Self-learning kinetic Monte Carlo method: Application to Cu(111), Phys. Rev. B. 72 (2005) 1154011–1154019.
[9] D. Konwar, V.J. Bhute, A. Chatterjee, An off-lattice, self-learning kinetic Monte Carlo method using local environments, J. Chem. Phys. 135 (2011) 174103.
[10] T. Rehman, M. Jaipal, A. Chatterjee, A cluster expansion model for predicting the activation barrier of atomic processes, J. Comp. Phys. 243 (2013) 244–259.
[11] N. Castin, L. Malerba, Calculation of proper energy barriers for atomistic kinetic Monte Carlo simulations on rigid lattice with chemical and strain field long-range effects using artificial neural networks, J. Chem. Phys. 132 (2010) 74507.
[12] E. Baibuz, S. Vigonski, J. Lahtinen, J. Zhao, V. Jansson, V. Zadin, F. Djurabekova, Migration barriers for surface diffusion on a rigid lattice: Challenges and solutions, Comput. Mater. Sci. 146 (2018) 287–302, https://doi.org/10.1016/j.commatsci.2017.12.054.
[13] K. Sastry, D.D. Johnson, D.E. Goldberg, P. Bellon, Genetic programming for multitimescale modeling, Phys. Rev. B. 72 (2005) 85438, https://doi.org/10.1103/PhysRevB.72.085438.
[14] L. Breiman, J. Friedman, R. Olshen, C. Stone, Classification and Regression Trees, Chapman & Hall/CRC, Wadsworth, Belmont, CA, 1984.
[15] J. Crank, The mathematics of diffusion, Second, Oxford University Press, Oxford, 1986.
[16] M.C. Tringides, ed., Surface Diffusion: Atomic and Collective Processes, Plenum Press, New York, 1997.
[17] G. Gilmer, Computer models of crystal growth, Science (80-.). 208 (1980) 355–363.
[18] M. Jagannath, S. Divi, A. Chatterjee, Kinetic Map For Destabilization Of Pt-Skin Au Nanoparticles Via Atomic Scale Rearrangements, J. Phys. Chem. C. 122 (2018) 26214–26225.
[19] P. Haldar, A. Chatterjee, Connectivity-list based characterization of 3D nanoporous structures formed via selective dissolution, Acta Mater. 127 (2017) 379–388, https://doi.org/10.1016/j.actamat.2017.01.049.
[20] P. Haldar, A. Chatterjee, Seeking kinetic pathways relevant to the structural evolution of metal nanoparticles, Model. Simul. Mater. Sci. Eng. 23 (2015) 025002, , https://doi.org/10.1088/0965-0393/23/2/025002.
[21] V. Imandi, M. Jagannath, A. Chatterjee, Role of solvent in metal-on-metal surface diffusion: A case for rational solvent selection for materials synthesis, Surf. Sci. 675 (2018) 54–63.
[22] V. Imandi, A. Chatterjee, Estimating Arrhenius parameters using temperature programmed molecular dynamics, J. Chem. Phys. 145 (2016) 034104, , https://doi.org/10.1063/1.4958834.
[23] C.L. Liu, J.M. Cohen, J.B. Adams, A.F. Voter, EAM study of surface self-diffusion of single adatoms of fee metals Ni, Cu, Al, Ag, Au, Pd, and Pt, Surf. Sci. 253 (1991) 334–344.
[24] J.M. Sanchez, F. Ducastelle, D. Gratias, Generalized cluster description of multicomponent systems, Physica A. 128 (1984) 334–350.
[25] A. Van der Ven, G. Ceder, M. Asta, P.D. Tepesch, A. Van der Ven, G. Ceder, First-principles theory of ionic diffusion with nondilute carriers, Phys. Rev. B. 64 (2001) 184307.
[26] S. Verma, T. Rehman, A. Chatterjee, A Cluster Expansion Model For Rate Constants Of Surface Diffusion Processes On Ag, Al, Cu, Ni, Pd And Pt (100) Surfaces, Surf. Sci. 613 (2013) 114–125.
[27] N. Kulkarni, A. Chatterjee, Capturing local atomic environment dependence of activation barriers in metals using cluster expansion models, J. Phys. Conf. Ser. 759 (2016) 012041.
[28] N. Castin, J.R. Fernández, R.C. Pasianot, Predicting vacancy migration energies in lattice-free environments using artificial neural networks, Comput. Mater. Sci. 84 (2014) 217–225, https://doi.org/10.1016/j.commatsci.2013.12.016.
[29] X.W. Zhou, R.A. Johnson, H.N.G. Wadley, Misfit-energy-increasing dislocations in vapor-deposited CoFe/NiFe multilayers, Phys. Rev. B. 69 (2004) 144113, , https://doi.org/10.1103/PhysRevB.69.144113.
[30] S. Divi, A. Chatterjee, Understanding Segregation Behavior in AuPt, NiPt, and AgAu Bimetallic Nanoparticles Using Distribution Coefficients, J. Phys. Chem. C. 120 (2016) 27296–27306, https://doi.org/10.1021/acs.jpcc.6b08325.
[31] S. Divi, G. Agrahari, S. Kadulkar, S. Kumar, A. Chatterjee, Improved Prediction Of Heat Of Mixing And Segregation In Metallic Alloys Using Tunable Mixing Rule For Embedded Atom Method, Model. Simul. Mater. Sci. Eng. 25 (2017) 085011.
[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in {P}ython, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[33] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Clarendon, 1995.