Electronic Supplementary Material (ESI) for Catalysis Science & Technology.

Supplementary Information for

Predicting Aggregation Energy for Single Atom Bimetallic Catalysts on Clean and O* Adsorbed Surfaces through Machine Learning Models

Zhuole Lu[a], Shwetank Yadav[a] and Chandra Veer Singh*[a,b]

[a]Department of Materials Science and Engineering, University of Toronto, 184 College St, Toronto, ON, Canada M5S 3E4

[b]Department of Mechanical and Industrial Engineering, University of Toronto

*Corresponding author

1. DFT surface model

   To ensure the (3×3) cell is sufficiently large, we conducted the same trimer-adsorbed $E_{ads,O*}$ calculations on (4×4) supercell models and compared with (3×3) supercell models. For this comparison, we randomly selected Al, Sc, Mn, Ga, Sr, Nb, Cd, Sb and Au as the dilute, which gave good coverage over all 38 elements considered in this study. The results from (3×3) and (4×4) supercells were essentially the same (Figure S1), with a fitted slope of 1.02 and a R^2 of 0.998. As this comparison was done on metal trimers, the largest cluster size, the results indicate that for all cluster sizes, the (3×3) supercells are likely large enough for the purpose of this study.

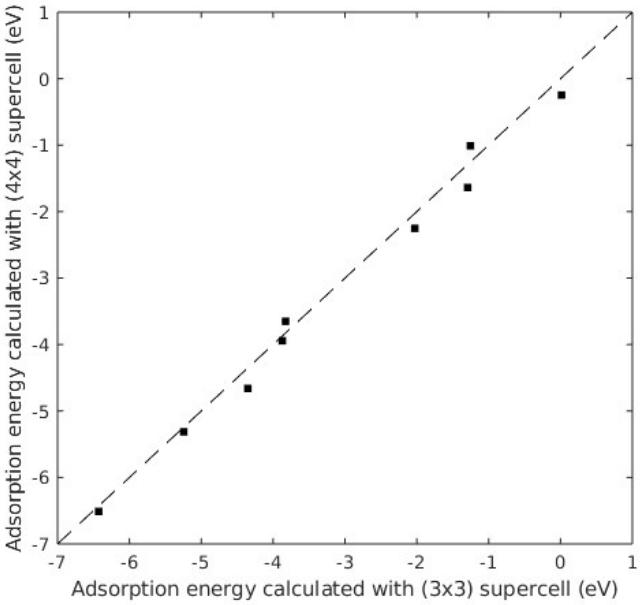
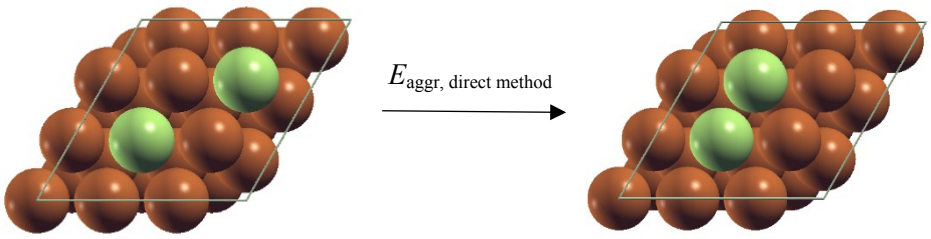


Figure S1. Parity plot of $E_{ads,O*}^{trimer}$ performed using (3×3) and (4×4) supercells.

2. Verification of methods for calculating aggregation energies

The method for calculating aggregation energies (eqs. 1-4 in the main text) is an indirect method, based on DFT-calculated total energies from separate cells. Although this method was used previously and verified carefully [1–3], we verified it again using a more direct method, using one single unit cell for DFT calculations (Figure S2).
Using the direct method, the aggregation energy is calculated as



$$E_{aggr}^{monomer \rightarrow dimer} = E(Cu - M - M) + E(Cu - Cu - M, M) \ (eq.S1)$$

Figure S2. Method for calculating the aggregation energy using the *direct method*, for verifying the *indirect method* used in the main text.

Comparing the direct method with the indirect method used in the main text (Figure S3), we find that the two yield generally the same results, thus having limited effect on our ML models. Also, since the direct method involves 2/9 = 0.22 coverage of M and the indirect method is 1/9 = 0.11 coverage of M (in the case of monomer), this also implies that the aggregation energy (at least for the monomer-to-dimer case) is invariant of the M coverage. In addition, this also verified that the (3×3) supercell is sufficiently large as two monomers within one (3×3) supercell has the same aggregation energy as two monomers from two separate supercells.

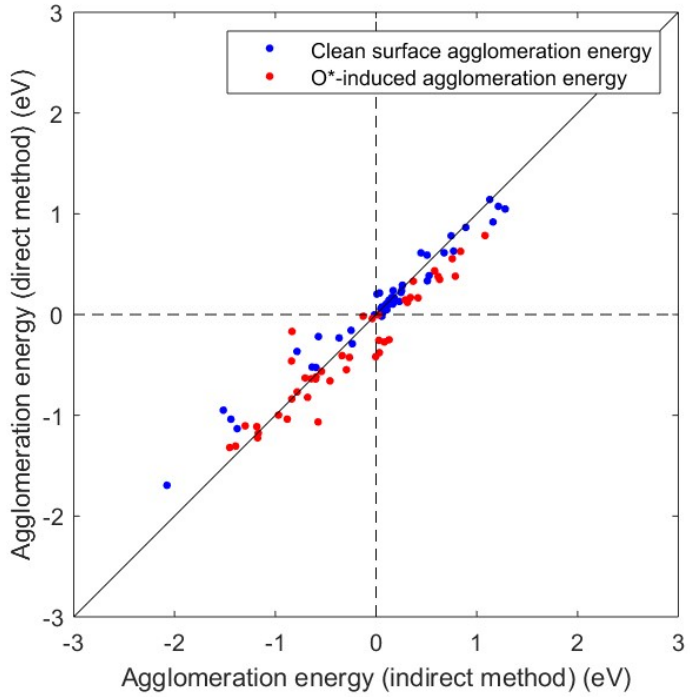


Figure S3. Parity plot for monomer-to-dimer aggregation energies calculated using the *direct* and *indirect* methods.

3. Details of implementation of ML algorithms and cross validation

All ML methods were implemented with MATLAB. ML training and prediction were based on built-in functions from the MATLAB machine learning package. For the pre-evaluation of ML methods, we employed the automatic Bayesian optimization algorithm to optimize important hyperparameters. The optimized hyperparameters are summarized in Table S1. Other parameters not explicitly listed are less important and are set to default values in MATLAB. We note that in the GPR algorithm, we set the lower bound of sigma to 0.1 eV, which signifies the uncertainty of the data points inherent with DFT. The value of 0.1 eV is a good approximation of DFT accuracy.[4]

For performance evaluation of ML models (Section 3.3 in main text), data were randomly shuffled and split into the training set (75% of the total data points) and testing set (the remaining 25% data points), and the ML models were only trained on the training set, with no knowledge of the testing set. We then predicted for the testing set and reported the prediction performance separately to ensure bias-free performance measures. We noted a strong effect of training/testing splitting on ML performance (Section 3.4 in main text), and accordingly Monte-Carlo cross-validation (i.e. repeated random sub-sampling validation) was performed, where data was randomly shuffled 200 times (subsequently split into 75% and 25% as training and testing data) and the performance metrics were averaged over all trials. Performance metrics

include the root mean squared error (RMSE) and mean absolute error (MAE). For results involving random shuffling, we set the random seed in MATLAB to "default" in order to ensure data reproducibility.

Table S1. Optimized hyperparameters of ML models during pre-evaluation

Model	Hyperparameter	Aggregation energy	Adsorption energy
GKR	Kernel scale	3.5798	10.082
	Lambda	9.8522e-5	2.6927e-5
SVM	Kernel scale	2.8292	2.9536
	Box constraint	61.376	205.56
	Epsilon	0.0043903	0.077687
GPR	Kernel function	ardmatern52	ardmatern52

4. Sensitivity analysis

We followed the perturbation method used in previous ML works,[5,6] where we perturbed each feature to +25% and normalized the corresponding change in the output:

$$\frac{1}{N}\sum_{i=1}^{N}\left|\frac{\partial E}{\partial f_i}\right|\frac{\max\left(f_i\right)-\min\left(f_i\right)}{\max\left(E\right)-\min\left(E\right)}$$

where N is the total number of data points, E is the output (aggregation or adsorption energy), and f_i is each feature. A greater response in the output indicates greater sensitivity in the corresponding feature.

5. Numerical data used for training the ML models

Tables S2 contains the numerical data of DFT results used for training the ML models.

Table S2. Numerical data for ML training

	Inputs					Outputs (eV)						
	AN	G	P	R (Angstrom)	EN	Aggr, Monomer to Dimer	Aggr, Dimer to Trimer	Aggr, Monomer to Dimer, with O*	Aggr, Dimer to Trimer, with O*	Ads, Mono-mer	Ads, Dimer	Ads, Trimer
Ca	20	2	4	1.94	1.04	0.890322	1.0279	-0.4572	-1.01813	-2.722	-4.069	-5.658
Sc	21	3	4	1.84	1.2	0.745071	1.0417	-0.96803	-1.3134	-3.327	-5.040	-6.427
Ti	22	4	4	1.76	1.32	0.008725	-0.012	-1.18377	-1.90574	-3.661	-4.853	-5.563
V	23	5	4	1.71	1.45	-0.596267	-0.038	-0.83698	-1.82215	-3.51	-3.754	-4.700
Cr	24	6	4	1.66	1.56	0.248274	0.4887	-0.67961	-1.09911	-2.840	-3.768	-4.676
Mn	25	7	4	1.61	1.6	0.18162	0.1645	-0.54207	-0.91804	-2.56	-3.285	-3.825
Fe	26	8	4	1.56	1.64	-0.366682	-0.470	-0.70398	-1.40473	-2.612	-2.949	-3.179
Co	27	9	4	1.52	1.7	-0.247839	-0.275	-0.5957	-1.12336	-2.438	-2.785	-3.037
Ni	28	10	4	1.49	1.75	-0.016374	-0.044	-0.33728	-0.64735	-2.036	-2.357	-2.622
Zn	30	12	4	1.42	1.66	0.046087	0.0921	-0.03724	-0.05351	-1.764	-1.847	-1.956

Ga	31	13	4	1.36	1.82	0.127692	0.1511	0.024294	0.076854	-1.825	-1.928	-2.027
Ge	32	14	4	1.25	2.02	0.16939	0.4547	0.36949	0.895124	-1.551	-1.350	-1.280
Se	34	16	4	1.03	2.48	0.107541	0.2760	0.028435	1.660713	-1.52	-1.60	-0.244
Sr	38	2	5	2.19	0.99	1.160955	1.3298	-0.00345	-0.09949	-2.65	-3.819	-5.245
Y	39	3	5	2.12	1.11	1.12862	1.3427	-0.88138	-0.80559	-3.221	-5.231	-6.498
Zr	40	4	5	2.06	1.22	0.446038	0.5361	-1.16872	-1.17724	-3.560	-5.175	-5.720
Nb	41	5	5	1.98	1.23	-0.784387	-0.456	-1.45195	-2.00064	-3.591	-4.258	-4.350
Mo	42	6	5	1.9	1.3	-2.075003	-0.72	-1.29838	-2.71229	-3.348	-2.572	-3.262
Tc	43	7	5	1.83	1.36	-1.378251	-0.500	-0.78325	-1.65423	-2.971	-2.376	-2.746
Ru	44	8	5	1.78	1.42	-0.235337	-0.05	-0.29401	-0.56482	-2.306	-2.36	-2.582
Rh	45	9	5	1.73	1.45	0.146286	0.4290	0.082216	0.13776	-1.772	-1.836	-2.20
Pd	46	10	5	1.69	1.35	0.059583	0.0843	0.341447	0.738098	-1.409	-1.12	-0.815
Ag	47	11	5	1.65	1.42	0.071526	0.1130	0.786881	1.379013	-1.617	-0.901	-0.422
Cd	48	12	5	1.61	1.46	0.101859	0.1266	0.29078	0.442769	-1.507	-1.318	-1.292
In	49	13	5	1.56	1.49	0.252666	0.5176	0.31151	0.549985	-1.549	-1.490	-1.769
Sn	50	14	5	1.45	1.72	0.526534	0.8709	0.615811	1.357451	-1.363	-1.27	-1.40
Sb	51	15	5	1.33	1.82	0.769528	1.1623	0.839069	1.73174	-1.052	-0.983	-1.253
Te	52	16	5	1.23	2.01	0.675248	1.1670	1.081587	2.234741	-0.936	-0.529	-0.543
Hf	72	4	6	2.08	1.23	0.506781	0.5526	-1.17527	-1.05639	-3.660	-5.342	-5.775
Ta	73	5	6	2	1.33	-0.572173	-0.469	-1.39388	-1.69179	-3.930	-4.752	-4.580
W	74	6	6	1.93	1.4	-1.515764	-1.444	-0.83842	-2.15349	-4.284	-3.606	-3.477
Re	75	7	6	1.88	1.46	-1.440478	-1.213	-0.83435	-1.9157	-3.28	-2.678	-2.546
Os	76	8	6	1.85	1.52	-0.634448	-0.243	-0.26425	-0.54131	-2.521	-2.150	-2.184
Ir	77	9	6	1.8	1.55	0.057534	0.2761	0.129691	0.311899	-1.82	-1.755	-1.848
Pt	78	10	6	1.77	1.44	0.096109	0.1463	0.416613	0.914178	-1.340	-1.020	-0.669
Au	79	11	6	1.74	1.42	0.166998	0.2857	0.756532	1.576949	-1.109	-0.520	0.0143
Tl	81	13	6	1.56	1.44	0.231191	0.307	0.581899	1.113117	-1.441	-1.091	-0.867
Pb	82	14	6	1.54	1.55	0.509197	0.7565	0.630741	1.524619	-1.342	-1.221	-1.083

6. Other figures

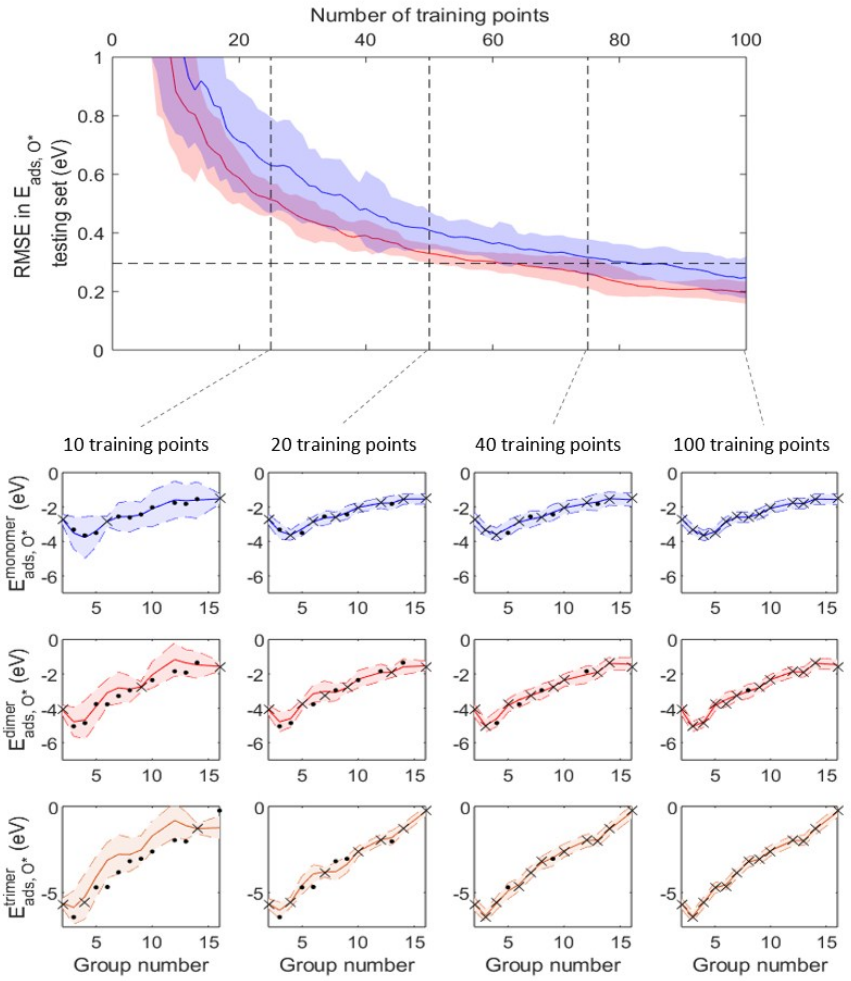


Figure S4. RMSE in predicted E_{ads,O^*}, as an increasing number of training points are selected using active learning algorithm (red) and random selection (blue). Solid line and shaded area are mean and ± 1σ of RMSE from 50 runs, each with a randomly chosen starting training point. Horizontal dotted line indicates average RMSE using 50 times of leave-25%-out (reported in Table 1) for better comparison. At each indicated stage is one ML model predicting for Period 4 elements. (× and · are training and testing set data respectively)

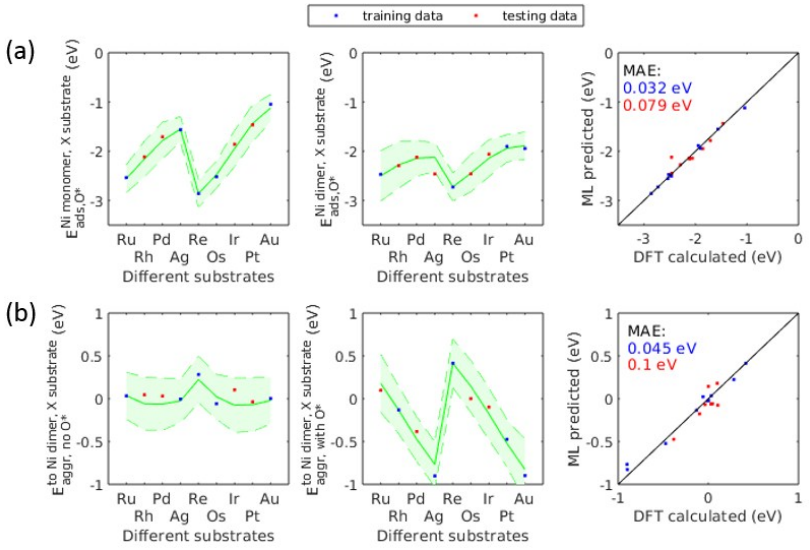


Figure S5. ML prediction on $E_{ads,\,O*}$ (a) and E_{aggr} (b) where the dilute secondary metal is Ni and substrates are varied, after the model is trained on 1/2 of the corresponding data set. Shaded areas represent 95% confidence interval. Parity plots of the same prediction are shown to the right, with MAE's on training (blue) and testing (red) data respectively.