Electronic Supplementary Material (ESI) for Journal of Materials Chemistry A.

Supporting Information

Machine Learning Scheme of Catalytic Activity of Alloys with Intrinsic Descriptors

Ze Yang, Wang Gao* and Qing Jiang*

School of Materials Science and Engineering, Jilin University, 130022 Changchun, P. R. China

Author Information

Corresponding Author

*wgao@mails.jlu.edu.cn

Note S1. Detailed information about tree ensemble models (RFR, ETR, GBR)

The tree ensemble methods evolved from the application of decision tree and ensemble method. Decision tree is a kind of non-parametric supervised learning method used for classification and regression. The goal of the method is to build a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. The advantage of decision tree is its easiness to understand and explain, due to its visible tree model. The disadvantage is that it is easy to generate over-complex trees that do not generalize the data well, which is called overfitting.[S1]

A single regression tree represents a simple piece-wise constant function, and an ensemble of multiple regression trees improves the flexibility of the model. This seems too simple for predicting continuous real values, but this simplicity is now widely known to work surprisingly well for general high-dimensional data.[S2] Another important advantage to use tree ensemble methods are that they are less dependant on hyperparameter settings, and thus even a ML amateur can obtain good prediction performance. In many practical cases, tree ensemble methods with default hyperparameters can give a good baseline, whereas kernel methods with default hyperparameters may give a very poor performance (even worse than constant prediction). To appropriately setup the hyperparameters of kernel methods (KRR, SVR, GPR), it would usually require some expertise and experience.[S3] Moreover, it can quantify the predictive power of a specific input feature by analyzing the reduction of the root mean squared error (RMSE) at each node of the tree to help us understand the importance of each feature.

The purpose of ensemble methods is to improve generalizability/robustness of a single estimator by combining the predictions of several base estimators, which are built with a given learning algorithm. Two families of ensemble methods are usually distinguished: (i) In averaging methods, the driving principle is to build several estimators independently and then to average their predictions and the combined estimator is usually better than any of the single base estimator because its variance is reduced; (ii) By contrast, in

boosting methods, base estimators are built sequentially and the bias of the combined estimator is reduced in each step.

For random forest regression (RFR), (i) n samples are selected from the data set using bootstrap sampling; (ii) k attributes are randomly selected from all attributes, and the best segmentation attribute is selected as the node to create a decision tree; (iii) m decision trees are established by repeating the above two steps for m times; (iv) these m decision trees form a random forest, and the voting result determines which range the data falls in.

For extra tree regression (ETR), as a variant of RFR, it has the basically same principle as RFR. Each sub-decision tree of the extra tree is trained with the original data set, and the ETR randomly selects one eigenvalue to segment the decision tree. This will result in a larger decision tree, and that is to say, the variance of the ETR model is further reduced relative to RFR. In some cases, the generalization ability of ETR is stronger than that of RFR.

For gradient boosting regression (GBR), it generates a weak prediction model based on the gradient of the loss function at each step, and weights them into the total model. GBR can simultaneously deal with continuous and discrete values by gradually reducing errors and has stronger robustness with less dependence on hyperparameter settings.

Note S2. Further discussion of ML algorithms

To reproduce the results in the reference, we first use the input features provided by the ref. 13 and build the model with kernel ridge regression (KRR) method in *scikit-learn* package in the same hyperparameter settings. As shown in Figure S4, although the errors (RMSE) we get are slightly higher than that in the reference, the trends of performance of the models are consistent with that in the reference regardless of whether the *d*-band center is excluded or included under 4,800 trials, which shows that the inclusion of the *d*-band center improves the accuracy significantly for all of the combinations of descriptors. This proves the reliability of our results, even though the sampling methods and the ML codes are different.

Next, we present the performance of our feature sets based on the models built by KRR algorithm instead of GBR algorithm in Figure S5. Slightly different from the GBR results, accuracy has a slight improvement (RMSE:0.35 eV→0.25 eV) with the inclusion of the *d*-band center, while the improvement is more significant (RMSE:0.45 eV→0.25 eV) in Figure S4. We infer that it is probably due to the different principles of the kernel methods and tree ensemble methods. The purpose of the kernel methods is to find and learn the mutual relationship in the original data. The original data is firstly embedded into the appropriate high-dimensional feature space by some kinds of nonlinear mapping, and then the general linear learner is used to analyze and process the patterns in this new space. However, our descriptors ψ_l and ψ in our feature sets are obtained by a simple analytical expression of the other two features, namely electronegativity and valence electron number. In other words, the kernel method needs more independent features to explore the correlation among features and to map to high-dimensional space in a way similar to our descriptor construction process. On the other hand, there is no such problem for the tree ensemble methods. As a base estimator of the tree ensemble methods, decision tree is a greedy algorithm strategy for segmentation based on entropy and it does not examine the correlation between each input feature.

Therefore, our descriptors perform better with tree ensemble methods than with kernel methods.

Furthermore, we have to point out that the usage of active learning method in the reference is not reasonable.[S4] In the case where the unlabeled data is abundant but manually labeling is expensive, the active learning algorithm can actively propose some labeling requests and submit some filtered data for experts tagging. The active learning method actually searches the entire data set first, and uses the most useful and different data in the data set as the training set. In one word, only in the face of massive unlabeled data, the active learning method should be used to reduce the training set and the labeling cost as much as possible. However, the data set used in the reference is less than 300 samples, and the model trained from the training set actively selected in the data set is still used in the original data set, so it can be foreseen that the performance of such a model is very good. Hence, although the performance of our model seems to be slightly lower than that in the reference, our model has stronger generalized predictive power and robustness because of the randomness of its sampling.

Figure S1 Results obtained from ML model based on the binding energy of various intermediates in ref. 4 including (a) CO*, (b) CHO* and (c) COH*.

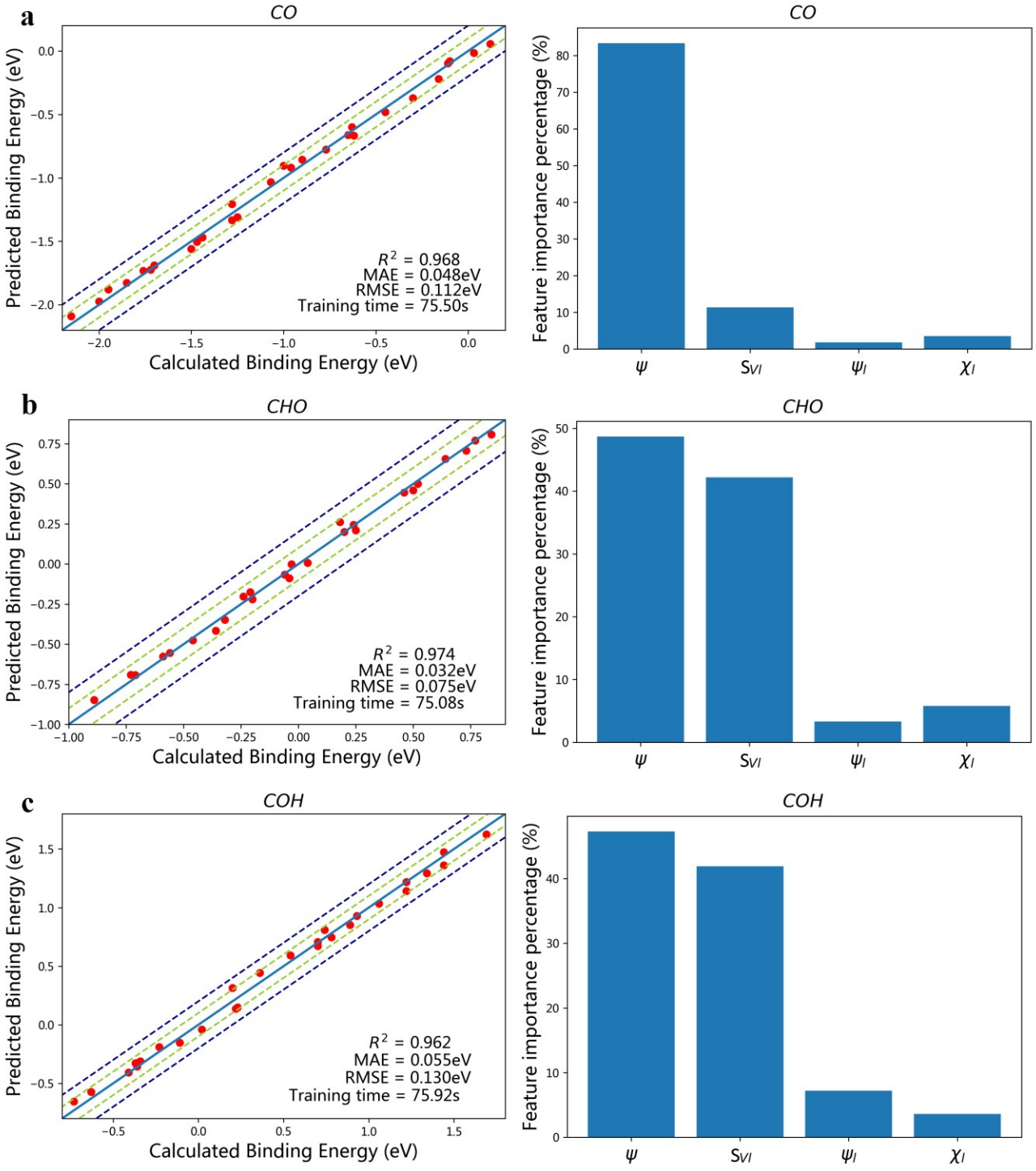

Figure S2 Results obtained from ML model based on the limiting potential of various reaction paths in ref. 4 including (a) CHO* and (b) COH*.

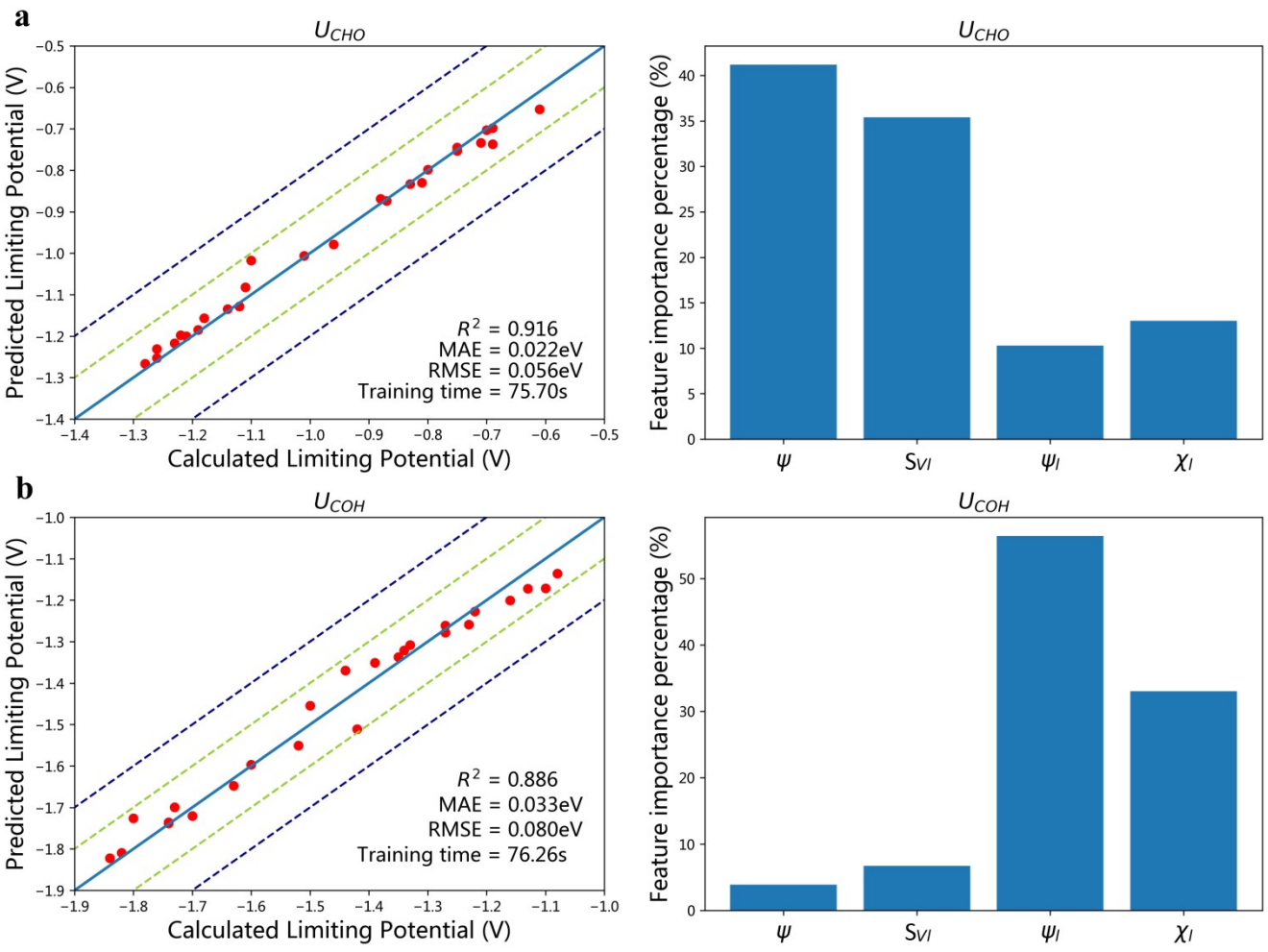

Figure S3 Results obtained from ML model based on the binding energy of various intermediates in ref. 5 including (a) C*, (b) CH*, (c) CH_2* and (d) CH_3*.

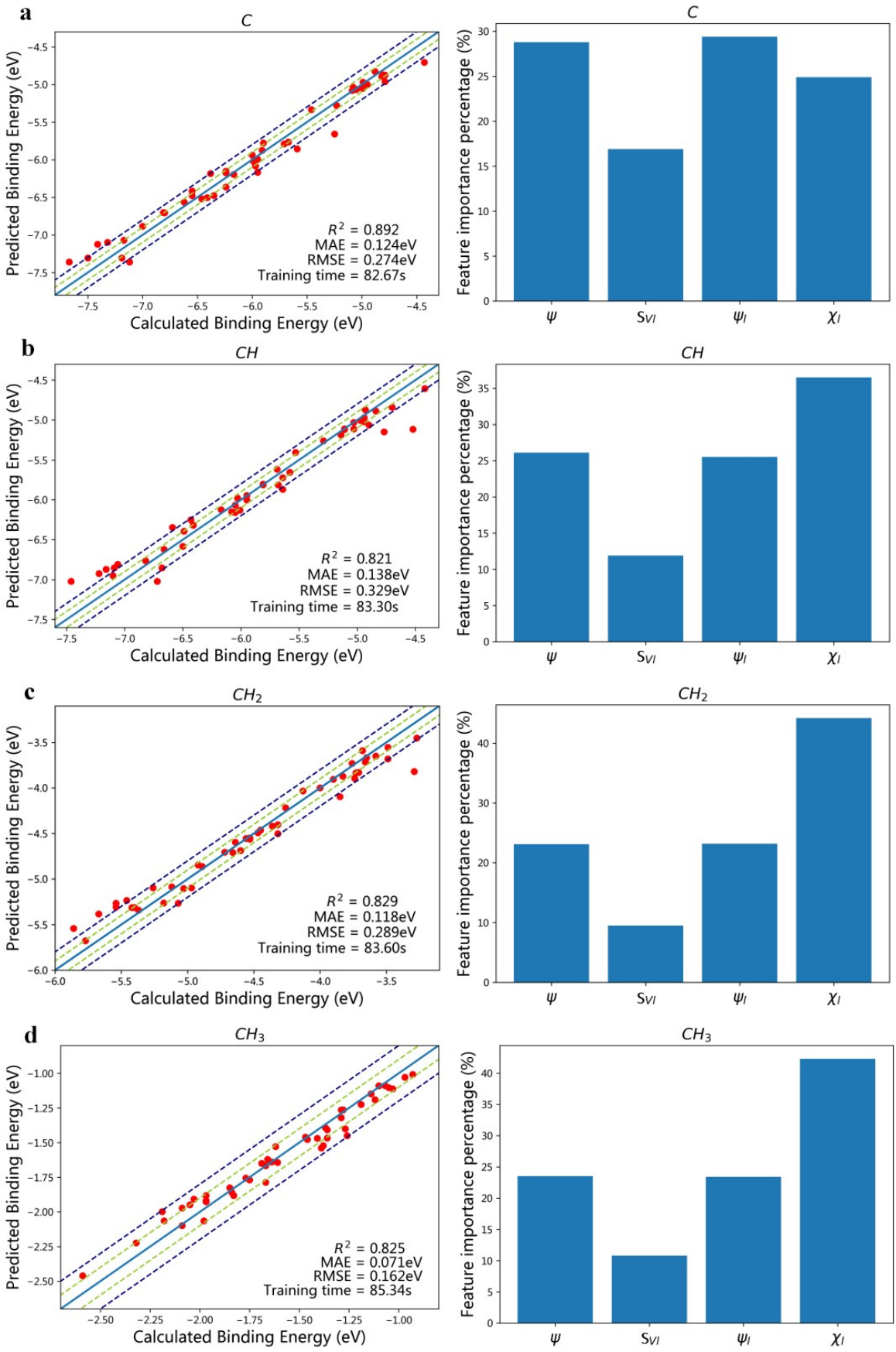
a
C
Predicted Binding Energy (eV)
Calculated Binding Energy (eV)
R^2 = 0.892
MAE = 0.124eV
RMSE = 0.274eV
Training time = 82.67s
C
Feature importance percentage (%)
ψ
S_{VI}
ψ_I
χ_I
b
CH
Predicted Binding Energy (eV)
Calculated Binding Energy (eV)
R^2 = 0.821
MAE = 0.138eV
RMSE = 0.329eV
Training time = 83.30s
CH
Feature importance percentage (%)
ψ
S_{VI}
ψ_I
χ_I
c
CH_2
Predicted Binding Energy (eV)
Calculated Binding Energy (eV)
R^2 = 0.829
MAE = 0.118eV
RMSE = 0.289eV
Training time = 83.60s
CH_2
Feature importance percentage (%)
ψ
S_{VI}
ψ_I
χ_I
d
CH_3
Predicted Binding Energy (eV)
Calculated Binding Energy (eV)
R^2 = 0.825
MAE = 0.071eV
RMSE = 0.162eV
Training time = 85.34s
CH_3
Feature importance percentage (%)
ψ
S_{VI}
ψ_I
χ_I

Figure S4. Performance of KRR algorithm with different feature sets taken from the reference: (a) without a *d*-band center, and (b) with a *d*-band center. All RMSE values were calculated for the whole data set.

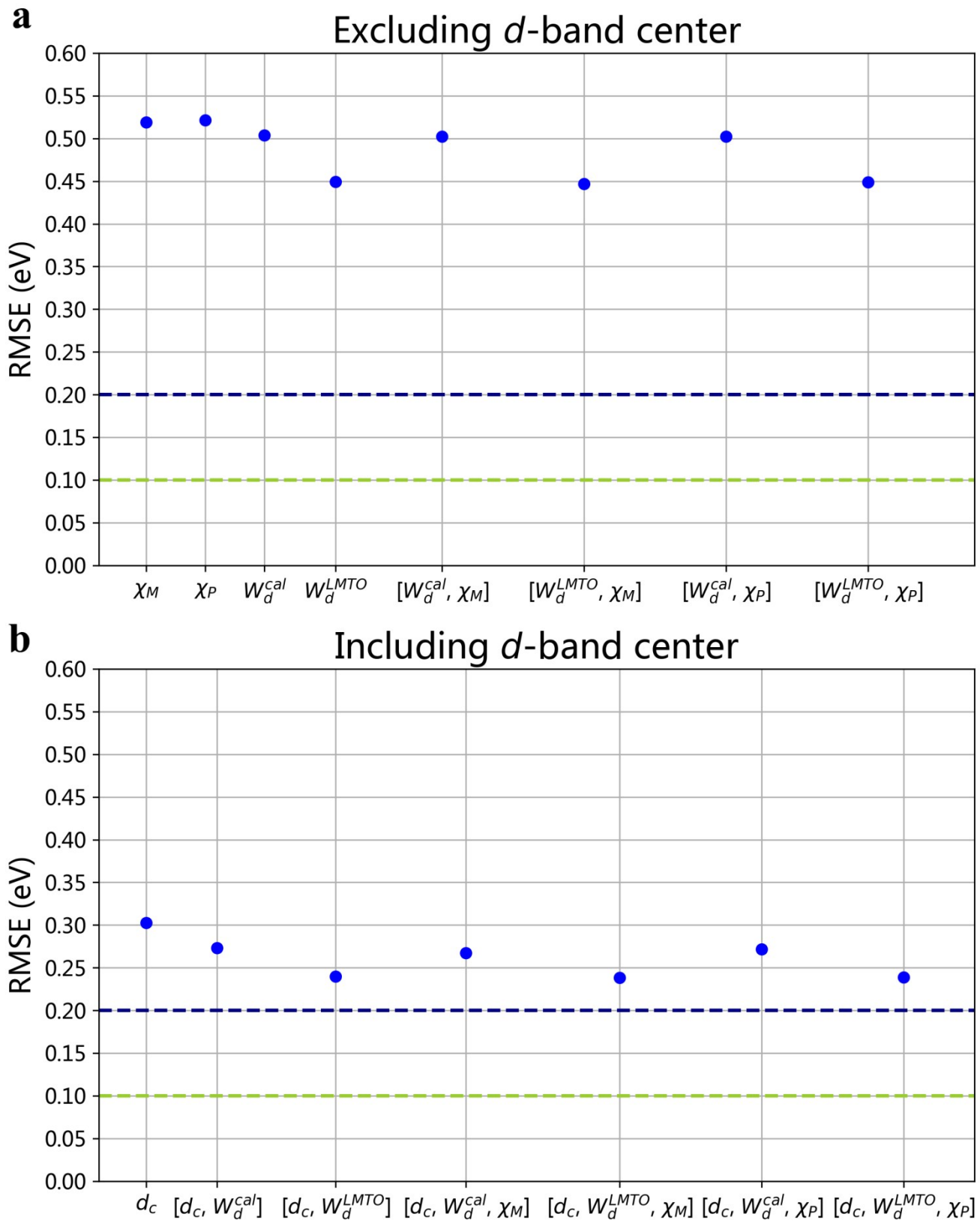

Figure S5. Performance of KRR algorithm with our different feature sets: (a) without a *d*-band center, and (b) with a *d*-band center. All RMSE values were calculated for the whole data set.

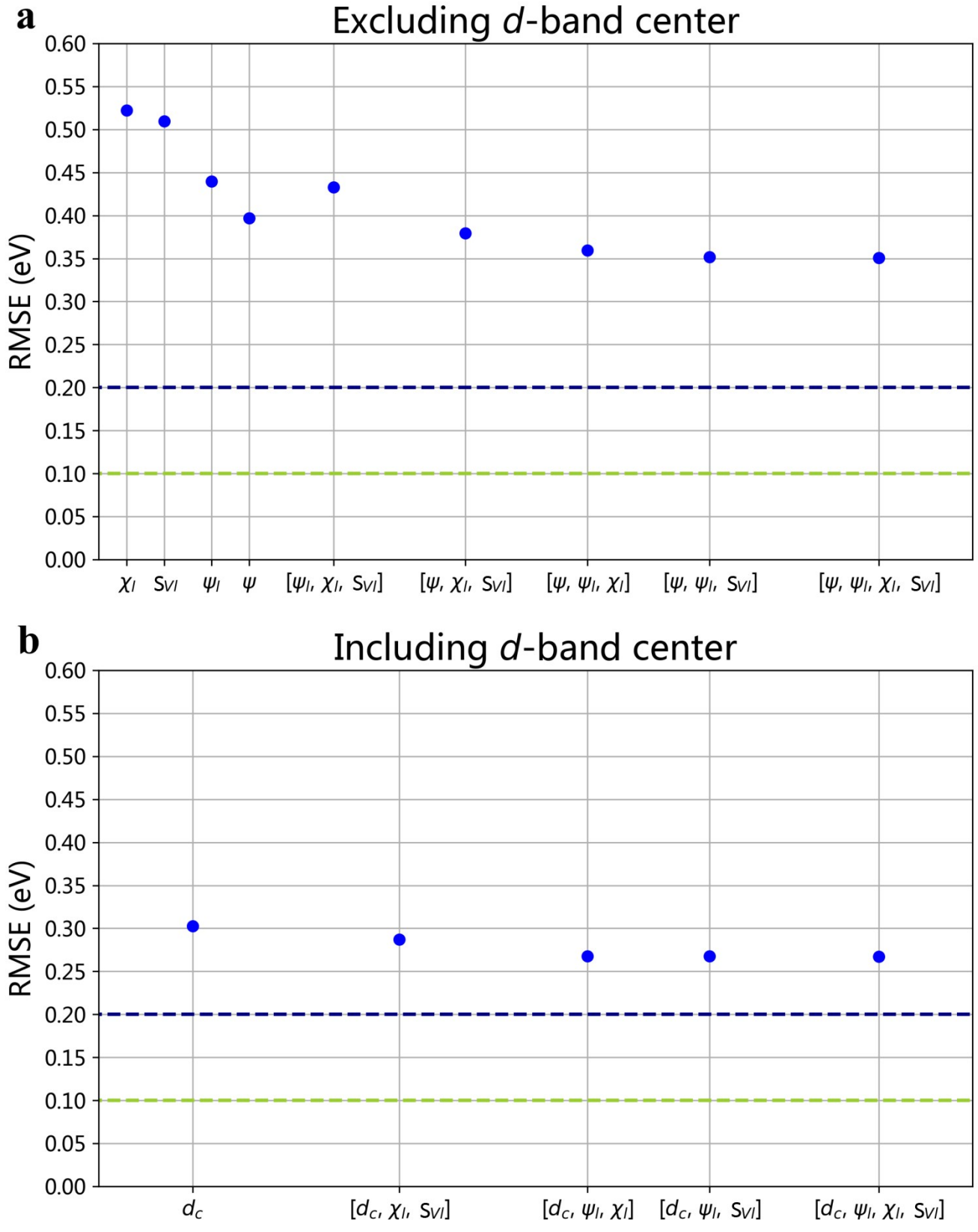

Table S1. Electronic and free energy corrections for gas phase species. All corrections were determined at 298.15 K. The electronic energies correction (E_{corr}) of CO were taken from the reference in order to adjust the limitation of PBE functional and ensure the consistency.

Species	E_{corr} (eV)	ZPE (eV)	-TS (eV)
CO(g)	-0.20	0.15	-0.61
H_2(g)	/	0.27	-0.40
H_2O(g)	/	0.56	-0.58

Table S2. Free energy corrections for adsorbed species. All corrections were calculated in this study and determined at 298.15K.

Species	ZPE (eV)	-TS (eV)
CO_2*	0.355	-0.175
COOH*	0.636	-0.122
HCOOH*	0.926	-0.168
CO*	0.179	-0.065
CHO*	0.490	-0.14
COH*	0.485	-0.122
CH_2O*	0.763	-0.149
CH_3O*	1.112	-0.157
CH_2OH*	1.111	-0.17
CH_3OH*	1.451	-0.148
CHOH*	0.799	-0.131
CH*	0.335	-0.049
CH_2*	0.669	-0.051

CH_3*	0.965	-0.065
CH_4*	1.283	-0.101
O*	0.076	-0.044
OH*	0.348	-0.093
H_2O*	0.622	0.17
H*	0.139	-0.066
H_2*	0.385	-0.068

Table S3. Performance of ML model based on all the combinations of input feature sets.

Input Features	R^2			R^2_{test}			MAE (eV)			RMSE (eV)		
	Mean	Max	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max	Min
χ_l	0.496	0.532	0.416	0.222	0.539	-0.368	0.292	0.31	0.278	0.372	0.401	0.359
S_{Vl}	0.736	0.748	0.689	0.646	0.784	0.298	0.233	0.246	0.226	0.269	0.292	0.263
ψ_l	0.699	0.752	0.593	0.338	0.739	-0.171	0.223	0.24	0.208	0.288	0.335	0.261
ψ	0.852	0.853	0.848	0.841	0.927	0.693	0.154	0.157	0.152	0.202	0.204	0.201
d_c	0.832	0.85	0.787	0.67	0.87	0.358	0.154	0.169	0.146	0.215	0.242	0.203
χ_l , S_{Vl}	0.816	0.848	0.763	0.601	0.817	0.298	0.173	0.19	0.16	0.225	0.255	0.205
χ_l , ψ_l	0.817	0.863	0.729	0.548	0.808	-0.013	0.166	0.186	0.149	0.224	0.273	0.194
χ_l , ψ	0.954	0.962	0.915	0.908	0.966	0.709	0.081	0.093	0.076	0.112	0.153	0.081
S_{Vl} , ψ_l	0.819	0.861	0.742	0.569	0.809	0.162	0.172	0.187	0.157	0.223	0.267	0.196
S_{Vl} , ψ	0.963	0.97	0.944	0.935	0.98	0.78	0.072	0.078	0.068	0.101	0.124	0.091
ψ_l , ψ	0.966	0.97	0.955	0.931	0.973	0.834	0.073	0.078	0.068	0.097	0.111	0.09
d_c , χ_l	0.953	0.966	0.926	0.869	0.965	0.673	0.076	0.086	0.069	0.113	0.143	0.096
d_c , S_{Vl}	0.928	0.942	0.9	0.83	0.928	0.646	0.099	0.111	0.09	0.141	0.166	0.127
d_c , ψ_l	0.944	0.961	0.922	0.849	0.936	0.605	0.084	0.093	0.075	0.123	0.147	0.104
d_c , ψ	0.931	0.943	0.907	0.847	0.942	0.659	0.098	0.107	0.092	0.138	0.16	0.125

χ_l , S_{Vl} , ψ_l	0.836	0.873	0.751	0.592	0.831	0.17	0.158	0.179	0.144	0.212	0.261	0.187
χ_l , S_{Vl}, ψ	0.978	0.984	0.961	0.947	0.987	0.82	0.054	0.063	0.049	0.077	0.104	0.067
χ_l , ψ_l , ψ	0.978	0.984	0.959	0.946	0.979	0.829	0.055	0.064	0.048	0.078	0.106	0.067
S_{Vl} , ψ_l , ψ	0.975	0.982	0.957	0.937	0.981	0.802	0.058	0.069	0.053	0.083	0.109	0.071
d_c , χ_l , S_{Vl}	0.958	0.972	0.938	0.88	0.966	0.726	0.071	0.08	0.064	0.107	0.131	0.088
d_c , χ_l , ψ_l	0.965	0.977	0.943	0.892	0.971	0.712	0.063	0.073	0.057	0.097	0.125	0.08
d_c , χ_l , ψ	0.976	0.984	0.959	0.929	0.976	0.819	0.055	0.064	0.049	0.081	0.106	0.066
d_c , S_{Vl} , ψ_l	0.956	0.97	0.928	0.87	0.96	0.641	0.074	0.084	0.066	0.11	0.141	0.091
d_c , S_{Vl} , ψ	0.977	0.984	0.963	0.94	0.982	0.839	0.057	0.065	0.051	0.08	0.101	0.066
d_c , ψ_l , ψ	0.977	0.985	0.958	0.933	0.982	0.82	0.054	0.06	0.047	0.079	0.108	0.063
χ , S_{Vl} , ψ_l , ψ	0.979	0.985	0.949	0.943	0.983	0.731	0.052	0.061	0.047	0.076	0.118	0.065
χ_l , S_{Vl} , ψ_l , d_c	0.965	0.977	0.944	0.889	0.961	0.723	0.063	0.073	0.056	0.098	0.124	0.079
χ_l , S_{Vl} , ψ, d_c	0.983	0.989	0.969	0.946	0.984	0.851	0.046	0.055	0.041	0.068	0.092	0.054
χ_l , ψ_l , ψ , d_c	0.984	0.991	0.963	0.946	0.985	0.813	0.043	0.053	0.037	0.066	0.1	0.05
S_{Vl} , ψ_l , ψ , d_c	0.983	0.99	0.967	0.945	0.985	0.828	0.044	0.052	0.039	0.068	0.095	0.051
χ_l , S_{Vl} , ψ_l , ψ , d_c	0.985	0.992	0.968	0.949	0.99	0.846	0.041	0.05	0.034	0.064	0.095	0.047

Table S4. Characteristic parameter α corresponding to various carbon-terminated intermediates involved in the carbon dioxide reduction reaction.

Species	$\boldsymbol{\alpha}$
C*	0.8
CH*	0.6

CH_2*	0.4
CH_3*	0.2
CO*	0.4
COH*	0.67
CHO*	0.3
CHOH*	0.47
COOH*	0.27
CH_2O*	0.1
CH_2OH*	0.27

References

S1. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel and B. Thirion, *Journal of Machine Learning Research*, 2011, **12**, 2825-2830.

S2. T. Chen and C. Guestrin, 2016.

S3. T. Toyao, K. Suzuki, S. Kikuchi, S. Takakusagi, K.-i. Shimizu and I. Takigawa, *The Journal of Physical Chemistry C*, 2018, **122**, 8315-8326.

S4. R. J. Brachman, W. W. Cohen and T. G. Dietterich, 2012.