Supporting Information

Machine Learning Assisted Computational Exploration of the Optimal Loading of IL in IL/COF Composites for Carbon Dioxide Capture

Tongan Yan,[ac] Minman Tong*[b], Dahuan Liu,*[a] Qingyuan Yang*[a] and Chongli Zhong[c]

[a]State Key Laboratory of Organic-Inorganic Composites, Beijing University of Chemical Technology, Beijing 100029, China

[b]Jiangsu Key Laboratory of Green Synthetic Chemistry for Functional Materials, Jiangsu Normal University, Xuzhou 221116, China

[c]State Key Laboratory of Separation Membranes and Membrane Processes, Tiangong University, Tianjin 300387, China

*Corresponding authors. E-mail addresses: tongmm@jsnu.edu.cn (M. Tong); liudh@mail.buct.edu.cn (D. Liu); qyyang@mail.buct.edu.cn (Q. Yang)

Table of Contents

S1. Potential Parameters and Point Charges of [MMIM][BF_4]

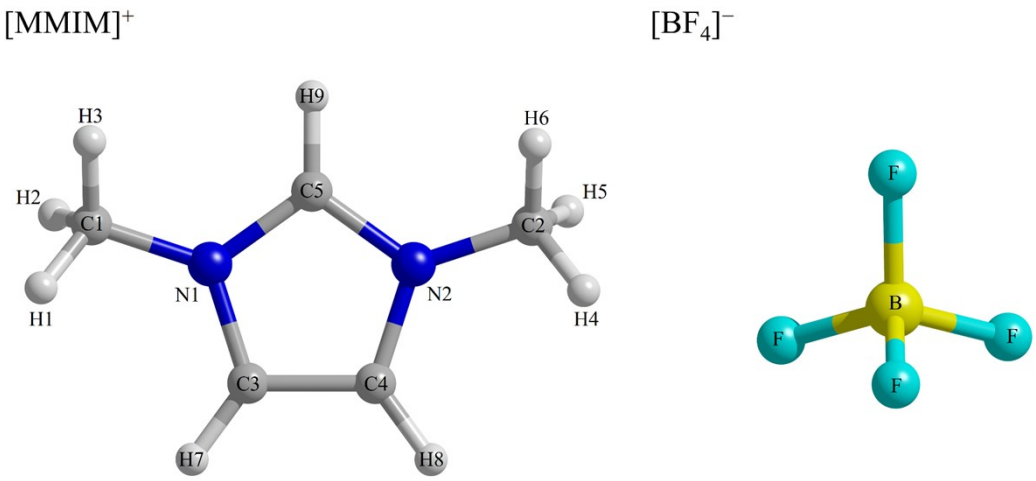

Fig. S1. Configuration of [MMIM][BF_4] and corresponding atomic IDs.

Table S1. Potential parameters[1,2] and point charges for [MMIM][BF_4].

Molecule	Atomic ID	Atomic type	ε/k_B (K)	σ (Å)	q (e)
[MMIM]+	N1	NA	85.55	3.250	0.173
	N2	NA	85.55	3.250	0.173
	C1	CT	55.05	3.400	0.384
	C2	CT	55.05	3.400	0.384
	C3	CW	43.28	3.400	−0.198
	C4	CW	43.28	3.400	−0.198
	C5	CR	43.28	3.400	0.144
	H1	H1	7.90	2.471	0.17
	H2	H1	7.90	2.471	0.17
	H3	H1	7.90	2.471	0.17
	H4	H1	7.90	2.471	0.17
	H5	H1	7.90	2.471	0.17
	H6	H1	7.90	2.471	0.17
	H7	H4	7.55	2.511	0.238
	H8	H4	7.55	2.511	0.238
	H9	H5	7.55	2.422	0.256
[BF_4]−	B	B	47.81	3.581	1.134
	F	F	30.70	3.118	−0.533

S2. Potential Parameters and Point Charges of Guest Molecules

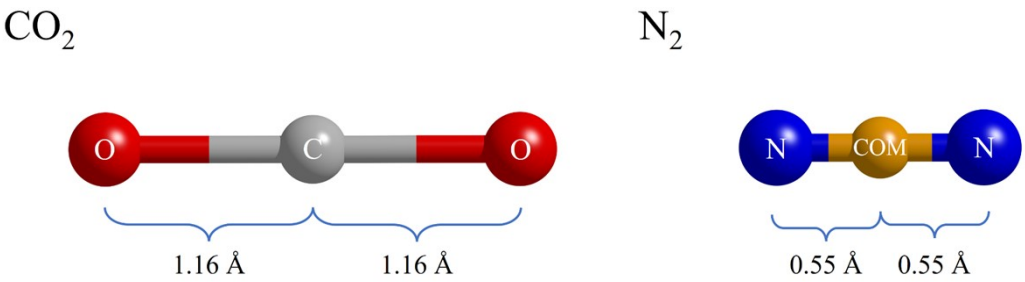

Fig. S2. Configuration of guest molecules and corresponding atomic types.

Table S2. Potential parameters[3] and point charges for CO_2 and N_2.

Molecule	Atomic type	ε/k_B (K)	σ (nm)	q (e)
CO_2	C	27.0	2.80	0.70
	O	79.0	3.05	−0.35
N_2	N	36.0	3.31	−0.48
	COM	0.0	0.0	0.96

S3. Machine Learning Algorithms

a) Categorical Boosting (CatBoost)

CatBoost is a new gradient boosting algorithm developed by Dorogush *et al.*,[4] which works successfully with categorical features by minimizing information loss. Compared to other gradient boosting algorithms, CatBoost is unique. To address the issue of target leakage, it first employs ordered boosting, an effective modification of gradient boosting algorithms. Additionally, this algorithm works well with small datasets. Third, CatBoost has the ability to manage categorical features. Typically, this handling is finished during the preprocessing stage and entails changing the original categorical variables to one or more numerical values.

b) eXtreme Gradient Boosting (XGBoost)

XGBoost model was developed by Chen *et al.*[5] and it is based on the gradient boosting decision tree. Compared with the traditional gradient boosting algorithm, the XGBoost model has made many improvements. It can be faster than other ensemble algorithms that use gradient boosting and have been considered an advanced evaluator with ultra-high performance in both classification and regression problems.

c) Comparison of CatBoost and XGBoost Algorithms

XGBoost has been widely used in successfully discovery and data analysis of nanoporous materials (e.g. MOFs) for gas storage and separation, as reflected from an excellent review[6]. In comparison, CatBoost is a relatively new algorithm but its applications in the field of separation-material design has not been fruitfully explored. Catboost has been demonstrated consistently faster for big-data mining than XGBoost. Furthermore, it has been demonstrated that CatBoost can obtain the best results in terms of generalization accuracy and AUC (Area Under Curve) than XGBoost.[7]

d) Optimization of ML Parameters

In machine learning, parameter tuning is a tedious but essential task, because it considerably affects the performance of the algorithm. Manual call-ups are time-consuming, and grid and random searches require no manpower but a long run time. In this paper, we use the Bayesian algorithm to adjust the parameters of our two machine learning models.

In this work, the Python package is used to train the CatBoost model. The Bayesian optimization method's process for modifying the four CatBoost model parameters for simulated data is described, respectively. And during that process, the primary parameters for tuning are learning_rate, max_depth, n_estimators, num_leaves, and boosting type. In Table S3**,** for simulated data of 7,746 [MMIM][BF_4]/COF composites based on 0−80 vol.%, the corresponding parameters are ultimately set to 0.8027, 16, 54, 35, and “gdbt”.

Table S3. Parameters optimization of CatBoost model by the Bayesian algorithm of 7,746 [MMIM][BF_4]/COF composites based on 0−80 vol.% with ILs loading ratio and geometric descriptors.

learning_rate	max_depth	n_estimators	num_leaves	R^2	
				Training set	Testing set
0.4229	12	10	16	0.864	0.830
0.1553	2	36	18	0.698	0.690
0.4028	9	69	35	0.965	0.940
0.2124	14	14	34	0.860	0.824
0.4231	9	30	11	0.928	0.903
0.8027	**16**	**54**	**35**	**0.990**	**0.949**
0.8776	14	22	3	0.966	0.925
0.1781	14	24	22	0.895	0.860
0.9583	9	107	16	0.982	0.943
0.6896	14	13	38	0.931	0.897
0.9890	12	49	40	0.983	0.938

0.1122	8	137	15	0.917	0.898
0.2949	3	13	34	0.667	0.662
0.2195	5	79	4	0.864	0.845
0.5784	3	93	35	0.905	0.883
0.1113	7	107	21	0.898	0.876
0.0595	9	103	26	0.875	0.853
0.9451	10	137	8	0.989	0.941
0.1479	13	66	9	0.946	0.918
0.9282	6	115	37	0.972	0.944

In the XGBoost model, there are five parameters, namely "gamma, learning_rate, max_depth, n_estimators, booster", which are focused during the parameter adjustment process. Using the Bayes Optimization method, we tune four parameters in a small range. The process of the Bayesian optimization method automatically adjusting the three parameters of the XGBoost model is shown in Table S4. Finally, the optimal parameters are set as follows: gamma = 3.974, learning_rate = 0.5434, max_depth = 7, n_estimators = 137, booster = "gbtree". Other parameters were the default values in the algorithm.

Table S4. Parameters optimization of XGBoost model by the Bayesian algorithm of 7,746 [MMIM][BF_4]/COF composites based on 0−80 vol.% with ILs loading ratio and geometric descriptors.

gamma	learning_rate	max_depth	n_estimators	R^2	
				Training set	Testing set
4.176	0.7231	1	61	0.701	0.705
1.476	0.1014	4	70	0.810	0.795
3.974	**0.5434**	**7**	**137**	**0.961**	**0.915**
2.052	0.8793	1	134	0.730	0.728
4.179	0.5631	3	40	0.892	0.868
8.009	0.9686	6	139	0.945	0.905
8.765	0.8957	2	9	0.714	0.728

1.707	0.8794	2	85	0.893	0.872
6.868	0.8363	1	150	0.731	0.729
9.889	0.7507	5	158	0.947	0.912
2.885	0.1387	1	136	0.675	0.683
2.124	0.2729	8	12	0.924	0.865
7.075	0.8232	10	135	0.963	0.8966
5.613	0.7382	10	32	0.964	0.900
3.032	1.0000	4	29	0.888	0.859
3.957	0.4606	7	30	0.960	0.916
4.435	0.2221	5	34	0.870	0.841

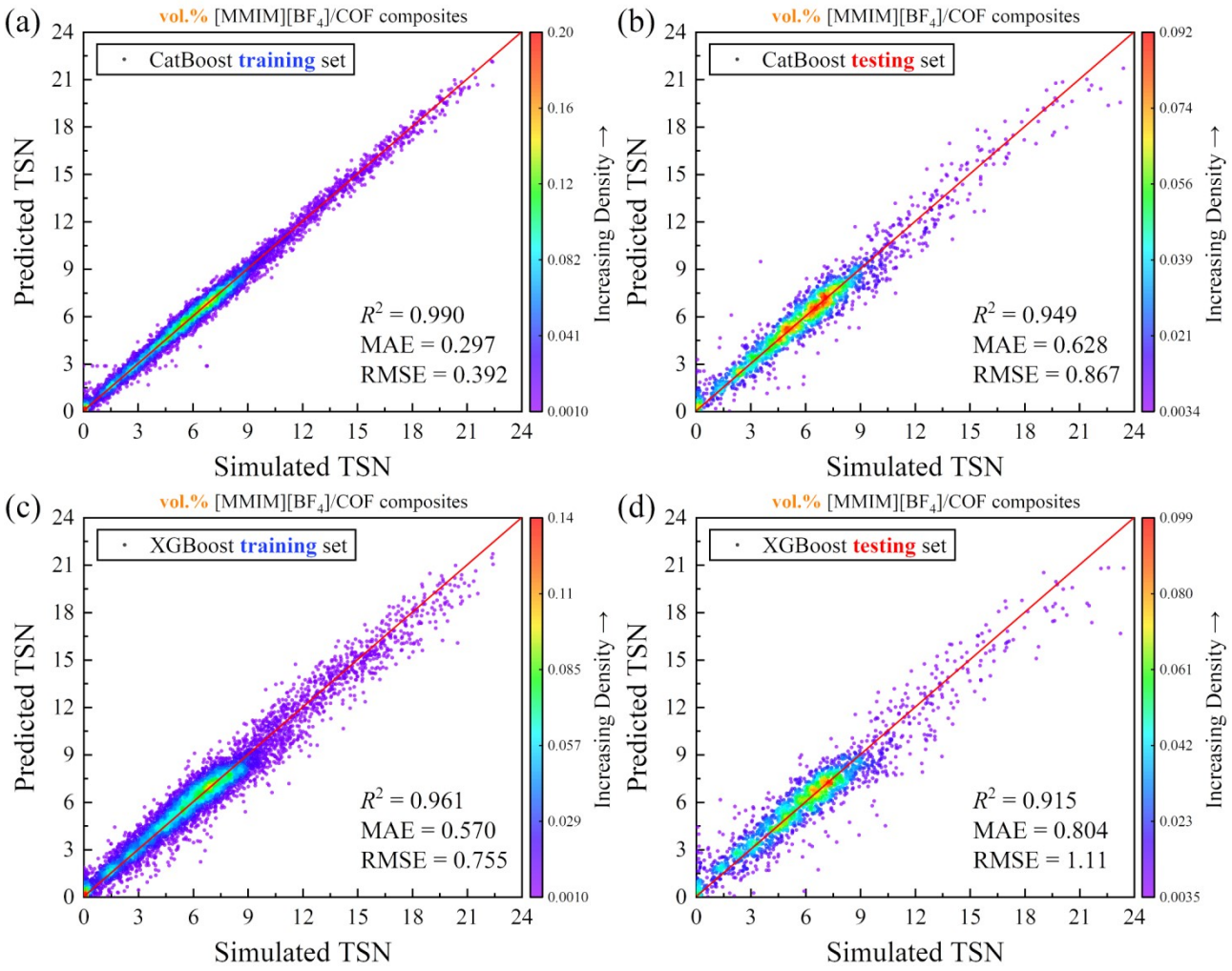

Fig. S3. Comparison of prediction results by two ML models with the GCMC-simulated TSN of 7,746 **vol.%-based** [MMIM][BF_4]/COF composites: (a) (b) CatBoost; (c) (d) XGBoost.

In Fig. S3, based on CatBoost and XGBoost models, the calculated and predicted TSN values of 7,746 vol.%-based [MMIM][BF_4]/COF composites are correlated. It can be seen that the data points are closely distributed along the diagonal (red line), and the R^2 value of both ML models exceeds ~0.96, along with very small MAE and RMSE values. This means that there is a good consistency between the calculated and predicted TSN values, indicating that the ML models are well-trained. This also proves that the feature descriptors we selected are sufficient for ML model training.

Table S5. Parameters optimization of **CatBoost model** by the Bayesian algorithm of 7,664 [MMIM][BF_4]/COF composites based on **0−80 wt.%** with ILs loading ratio and geometric descriptors.

learning_rate	max_depth	n_estimators	num_leaves	R^2	
				Trainingset	Testingset
0.4229	12	1	31	0.417	0.412
0.1553	2	38	35	0.650	0.645
0.4028	**9**	**84**	**69**	**0.967**	**0.942**
0.2124	14	6	67	0.707	0.695
0.4231	9	29	21	0.911	0.884
0.8040	2	140	11	0.875	0.855
0.8776	14	18	5	0.951	0.914
0.1781	14	21	43	0.870	0.847
0.0100	1	71	78	0.198	0.214
0.6896	14	5	75	0.822	0.796
0.0100	1	163	35	0.299	0.309
0.1122	8	182	30	0.927	0.906
0.2949	3	5	68	0.383	0.384
0.2195	5	99	6	0.874	0.859
0.5784	3	118	70	0.906	0.885
0.1113	7	139	42	0.908	0.892
0.0595	9	133	52	0.883	0.864
0.2070	2	101	41	0.588	0.575
0.1479	13	80	17	0.948	0.925
0.1781	14	21	43	0.869	0.840

Table S6. Parameters optimization of **XGBoost model** by the Bayesian algorithm of 7,664 [MMIM][BF_4]/COF composites based on **0−80 wt.%** with ILs loading ratio and geometric descriptors.

gamma	learning_rate	max_depth	n_estimators	R^2	
				Training set	Testing set
4.176	0.7231	1	61	0.607	0.588
1.476	0.1014	4	70	0.781	0.764
2.052	0.8793	1	134	0.638	0.604
4.179	0.5631	3	40	0.881	0.858
8.009	0.9686	6	139	0.934	0.898
8.765	0.8957	2	9	0.680	0.655
1.707	0.8794	2	85	0.892	0.865
6.868	0.8363	1	150	0.642	0.612
9.889	0.7507	5	158	0.934	0.896
2.885	0.1387	1	136	0.590	0.577
2.124	0.2729	8	12	0.927	0.882
3.974	**0.5434**	**7**	**137**	**0.964**	**0.931**
7.677	0.826	10	134	0.957	0.896
7.732	0.7915	8	32	0.950	0.903
6.552	0.4765	8	99	0.955	0.920
0.010	1.0000	3	99	0.898	0.870
7.299	1.0000	8	84	0.959	0.915

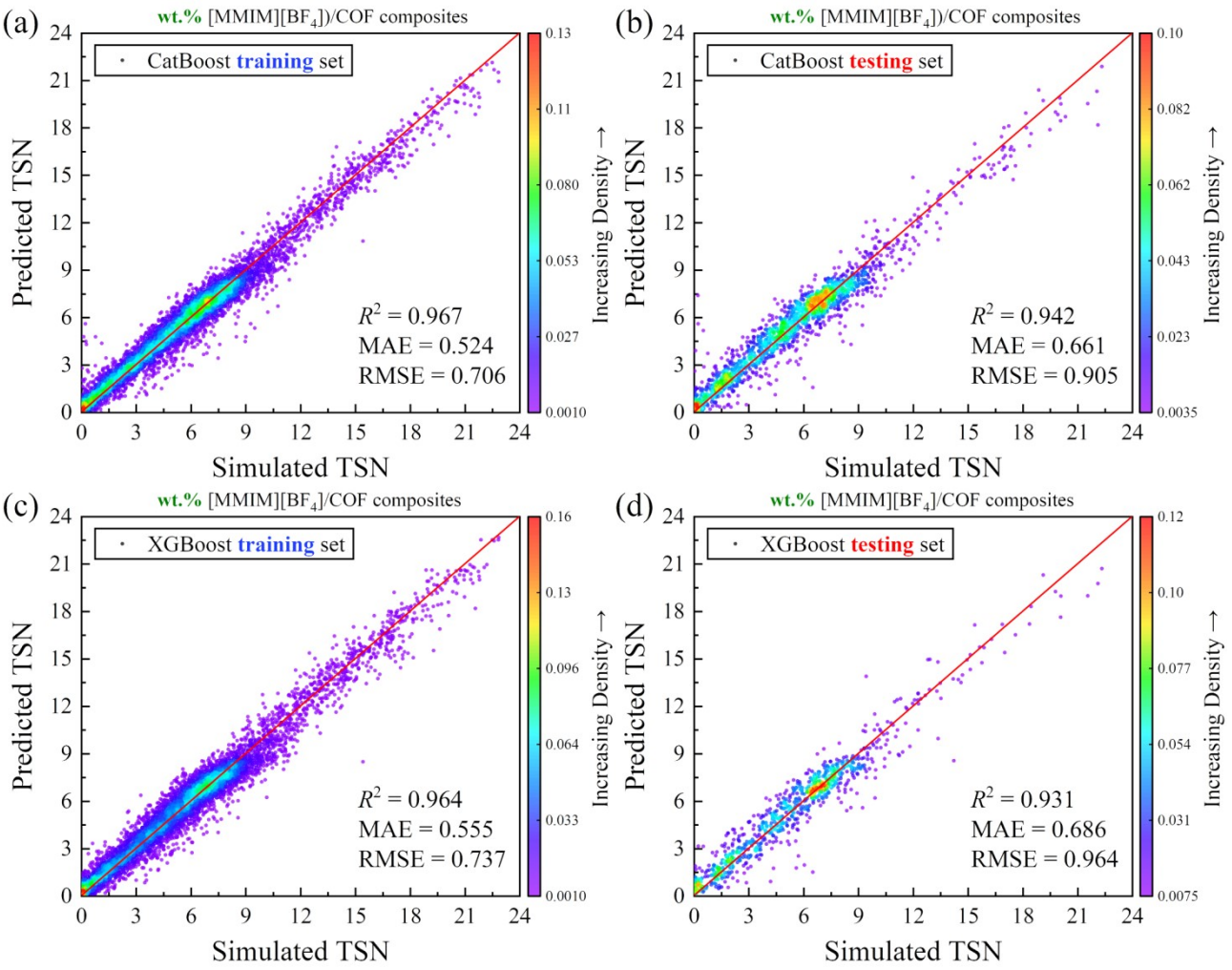

Fig. S4. Comparison of prediction results by two ML models with the GCMC-simulated TSN of 7664 **wt.%-based** [MMIM][BF_4]/COF composites: (a) (b) CatBoost; (c) (d) XGBoost.

In Fig. S4, both the CatBoost and XGBoost models are well-trained according to simulation data of 7746 wt.%-based composites, the R^2 value of both ML models exceeds ~0.96. In the training process of ML models, two groups of simulation data adopt the same feature descriptors, which means that the effect of feature descriptors on the separation performance of composites can be compared more intuitively. Considering that the accuracy of the CatBoost model is slightly higher than that of XGBoost, the following analysis aims at the output results of CatBoost.

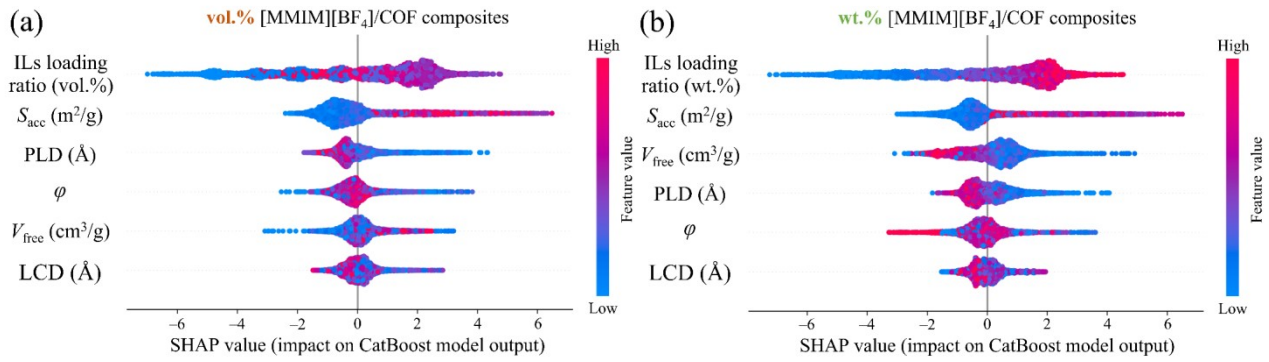

Fig. S5. Local interpretation (SHAP value distribution) of [MMIM][BF_4]/COF composites by the CatBoost model: (a) 7,746 vol.%-based; (b) 7,664 wt.%-based. (Red points: higher descriptor values, Blue points: lower descriptor values; Wide: dense distribution of samples, Narrow: sparse distribution of samples)

The SHAP values are calculated to quantify the effect of features on the CatBoost model output in terms of both magnitudes (significant or insignificant) and direction (positive or negative). Also, the color represents the value of the feature and is scaled to the same range. Fig. S5 shows the distribution of SHAP values for all composites, with the 6 descriptors listed in descending order of importance. The results show that no matter vol.% or wt.% is used for composite construction, the parameter of ILs loading ratio has the greatest impact on the output of the CatBoost model.

References

1. W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell and P. A. Kollman, A second generation force field for the simulation of proteins, nucleic acids, and organic molecules, *J. Am. Chem. Soc.*, 1995, **117**, 5179–5197.
2. K. M. Gupta, Y. Chen, Z. Hu and J. Jiang, Metal–organic framework supported ionic liquid membranes for CO_2 capture: anion effects, *Phys. Chem. Chem. Phys.*, 2012, **14**, 5785–5794.
3. M. G. Martin and J. I. Siepmann, Transferable potentials for phase equilibria. 1. United-atom description of n-alkanes, *J. Phys. Chem. B*, 1998, **102**, 2569–2577.
4. T. Q. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, *Kdd'16: Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining 2016*, pp. 785–794.
5. A. V. Dorogush, V. Ershov and A. Gulin, CatBoost: gradient boosting with categorical features support, 2018, arXiv:1810.11363. arXiv.org e-Print archive. https://arxiv.org/abs/1810.11363.
6. J. Lin, Z. Liu, , Y. Guo, S. Wang, Z. Tao, X. Xue, R. Li, S. Feng, L. Wang, J. Liu, H. Gao, G. Wang and Y. Su, Machine learning accelerates the investigation of targeted MOFs: Performance prediction, rational design and intelligent synthesis, *Nano Today*, 2023, **49**, 101802.
7. C. Bentéjac, A. Csörgő and G. Martínez-Muñoz, A comparative analysis of gradient boosting algorithms, *Artif. Intell. Rev.*, 2021, **54**, 1937–1967.