Accelerating the Selection of Covalent Organic Frameworks with Automated Machine Learning

Peisong Yang, Huan Zhang, Xin Lai, Kunfeng Wang, Qingyuan Yang,* and Duli Yu

Cite This: ACS Omega 2021, 6, 17149−17161

ABSTRACT: Covalent organic frameworks (COFs) have the advantages of high thermal stability and large specific surface and have great application prospects in the fields of gas storage and catalysis. This article mainly focuses on COFs’ working capacity of methane (CH₄). Due to the vast number of possible COF structures, it is time-consuming to use traditional calculation methods to find suitable materials, so it is important to apply appropriate machine learning (ML) algorithms to build accurate prediction models. A major obstacle for the use of ML algorithms is that the performance of an algorithm may be affected by many design decisions. Finding appropriate algorithm and model parameters is quite a challenge for nonprofessionals. In this work, we use automated machine learning (AutoML) to analyze the working capacity of CH₄ based on 403,959 COFs. We explore the relationship between 23 features such as the structure, chemical characteristics, atom types of COFs, and the working capacity. Then, the tree-based pipeline optimization tool (TPOT) in AutoML and the traditional ML methods including multiple linear regression, support vector machine, decision tree, and random forest that manually set model parameters are compared. It is found that the TPOT can not only save complex data preprocessing and model parameter tuning but also show higher performance than traditional ML models. Compared with traditional grand canonical Monte Carlo simulations, it can save a lot of time. AutoML has broken through the limitations of professionals so that researchers in nonprofessional fields can realize automatic parameter configuration for experiments to obtain highly accurate and easy-to-understand results, which is of great significance for material screening.

1. INTRODUCTION

Natural gas is an important high-quality clean energy. The main component of natural gas is alkanes, of which methane (CH₄) accounts for the vast majority. Because natural gas has a very low carbon/hydrogen ratio, it can avoid many pollution problems. The advantages of CH₄’s high calorific value, large reserves, and wide distribution make it a gradual alternative to oily fuels, gradually leading to a global energy transformation.¹² The main obstacle to restrict the use of natural gas in fuel comes from its storage problem. The US Department of Energy (DOE) set a storage target for CH₄ in 2012: at room temperature and 65 bar, it can compress and store 315 cubic centimeters of CH₄ to 1 cm³. However, the storage capacity of existing materials is still far below this goal.³ In order to meet the storage requirements for CH₄, it is imperative to develop and design high-performance solid adsorbents. Molecular sieves are the earliest materials used for adsorption storage, but due to their low surface area and strong hydrophilicity, their storage capacity is poor.³ In recent years, the emerging materials, covalent organic frameworks (COFs), have become a hot research topic. Due to their high specific surface area, low density, high thermal stability,⁵ they have been widely used in the fields of gas storage, separation,
the formula intuitively. A support vector machine (SVM) can avoid the "curse of dimensionality" to some extent. A decision tree (DT) needs a relatively small amount of calculation and can be easily converted into classification rules. A random forest (RF) performs well for many data sets and has relatively high accuracy. Although some traditional ML algorithms such as MLR, SVM, DT, and RF can solve many classification and regression problems, there are still many shortcomings. With the increase in the types of algorithms and problem complexity, researchers need to carefully choose the appropriate model architecture, training process, regularization method, and hyperparameters, all of which have great impact on the final performance of the algorithm. The process of building an accurate and powerful ML model requires advanced data processing and analysis skills. Choosing an appropriate method to solve the problem and configure optimal parameter values for a specific model is quite difficult. In order to solve the above problems, AutoML has become a research hotspot.

AutoML automates some common steps to simplify the process of generating models in ML. It aims to combine data preprocessing, feature engineering, algorithm selection, and model parameter optimization and put them in a black box. We only need to input data and wait for the data processing and prediction results. AutoML can integrate the iterative process of traditional ML. It can automatically search for the best combination of model parameters and realize the functions of automatic feature engineering, pipeline matching, parameter adjustment, and model selection, thereby reducing the waste of time and labor resources. AutoML not only solves the bottleneck of the talent gap in the AI industry but also lowers the threshold of ML and makes AI more widely used.

In this paper, we use AutoML to build a prediction model of the structure–effect relationship for the COFs database. First, we calculate 23 characteristics of 403,959 COFs and the working capacity of CH₄ through GCMC simulations and then use the tree-based pipeline optimization tool (TPOT), which is a method for rapid model selection and parameter adjustment based on the evolutionary algorithm (EA) to automatically configure the pipeline and the best model parameters for the data set and finally obtain prediction accuracy up to 99%. Compared with traditional ML models, the TPOT model has higher accuracy and saves a lot of time in model learning.

The main contributions of the paper are threefold:

First, the ML methods are used to study 403,959 COFs, and the specific features affecting material adsorption are obtained, which provides a theoretical basis for future material design.

Second, the use of AutoML method to build prediction models is much more efficient than previous manual selection of ML algorithms and hyperparameters. The parameter adjustment process is more scientific and interpretable, and accurate prediction models can be obtained more quickly.

Third, this work presents a new analytical idea so that nonprofessionals can easily use ML methods for data analysis in their specific field. The application of AutoML to materials science has lowered the threshold of AI application, which provides interdisciplinary researchers with valuable inspiration. It shortens the calculation time, improves the screening efficiency, and reduces the design cycle of materials.

The remainder of this article is organized as follows. Section 2 introduces some related works, including the combination of ML and materials, and the development and application of AutoML. Section 3 describes the overall framework of this work and presents the data source of materials and the computational methods. The experimental results are described in Section 4. Finally, the conclusion is drawn in Section 5.

2. RELATED WORKS

The application of ML in materials science is becoming more and more extensive. In many previous works, material researchers are keen to combine material research with ML. ML methods have become powerful tools for complex data analysis and information mining in materials science. The algorithm model of ML can quickly and accurately predict the performance indicators. Compared with the traditional calculation method of adsorption amount, it can save a lot of calculation time and has been successfully applied in materials science. Fanourgakis et al. introduce chemical features in descriptors by using the “type” of the atoms in MOFs to account for the chemical character of MOF. Kim et al. propose a novel Monte Carlo-machine learning (MC-ML) strategy to predict methane isotherms at a range of temperatures from a methane isotherm at 298 K. Gülsoy et al. extract knowledge on CH₄ delivery capacities (from 65 to 5 bar) of MOFs by using DT and neural networks as data mining tools. They use six user-defined descriptors and analyze two structure attributes with an artificial neural network and DT and obtain a simple and effective path for screening MOF materials. All these show that ML plays an important role in identification and screening of high-performance materials, and many researchers use data mining to study the prediction of material properties. According to the structural chemical characteristics of the material, its adsorption performance can be accurately predicted, and impressive results have been achieved. However, getting an accurate prediction model is difficult for many researchers. The generation of the model needs to go through complex feature engineering, algorithm selection, hyperparameter optimization, model training, and model evaluation. Every step is tediously debugged by engineers. The process takes a lot of time and is expensive, leading to the upsurge of AutoML.

AutoML is an emerging research direction in recent years. It aims to provide nonexperts with access to ML and enable experts to focus on other aspects of the ML process, rather than pipeline design. At present, the research of AutoML mainly focuses on the model selection and parameter optimization, that is, selecting appropriate algorithm model according to the data set and configuring the parameters with the best performance. There are mainly four parameter search methods: grid search, random search, Bayesian optimization, and EA optimization. Due to the huge computation and high cost of grid search and random search, the combination of hyperparameters for models exploring high-dimensional space (more than 10 hyperparameters) is impractical. Therefore, Bayesian optimization and EA optimization search are frequently used in practice. Both auto-sklearn and auto-WEKA use the Bayesian optimization method. Although both systems allow the use of simple ML pipelines, including data preprocessing, feature engineering, and single model prediction, they cannot build more complex pipelines or stacked models, which are necessary for complex prediction problems. The TPOT is a tool for optimizing ML pipelines based on the EA. It can generate highly scalable and complex ML pipelines and integrated models for data scientists. The EA has a high
degree of flexibility when gradually constructing pipelines in complex ML algorithms and their hyperparameter search spaces.24 Tsamardinos et al.25 use AutoML to automatically generate prediction models in the MOF adsorption research. Compared with other ML methods, they can obtain better prediction results and correctly predict the performance of MOFs.

In this paper, we use the TPOT tool in AutoML. Without prior knowledge, the TPOT has proven superior to traditional ML methods. The TPOT is used for automatic modeling of ML algorithms, which solves the difficulty of parameter adjustment in the application of ML. Finding the best algorithm model and parameter configuration among tens of thousands of model combinations is better than artificial settings. The accuracy and efficiency of the model obtained by adjusting the parameters have been greatly improved, which plays an important role in the selection and design of materials.

3. METHODS FOR MATERIALS DATA GENERATION AND ANALYSIS

3.1. Feature Engineering. Feature engineering is an important concept in the ML field. It is the work of designing feature sets for ML applications, focusing on how to design features that match the characteristics of the data itself and the application scenario. Feature engineering usually includes feature extraction, feature selection, and feature construction.26 Feature extraction usually aims to reduce the dimension of features through some functional mapping, while feature construction is used to expand the original feature space. In addition, the purpose of feature selection is to reduce feature redundancy by selecting important features. Feature selection is the process of constructing feature subsets through the reduction of unrelated or redundant features from the original feature set, which is beneficial to simplify the model, thereby avoiding overfitting and improving model performance. Feature construction is the process of constructing new features from the basic feature space or original data to enhance the accuracy and generalization of the model. Its essence is to improve the representation of the original features. Traditionally, this process highly depends on human expertise, while the commonly used method is data preprocessing. Feature extraction is a dimension reduction process via certain mapping functions. This process extracts highly informative and nonredundant features according to certain specific indicators. Unlike feature selection, feature extraction usually changes the original feature.

For feature extraction and construction, we use the principal component analysis (PCA) method, which is a multivariate statistical analysis method.27 This method forms new variables by constructing a series of linear combinations of original variables so that the new variables reflect as much information of the original variables as possible on the premise that they are not related to each other. Data information is mainly reflected in the variance of data variables. The larger the variance, the more information it contains. It is usually measured by the cumulative variance contribution rate. First, perform central standardization processing to generate a standard matrix

$$X_{N \times m} = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nm} \end{bmatrix}$$

(1)

$$x^*_i = \frac{(x_i - \bar{x})}{\sigma}$$

(2)

where \(X_{N \times m}\) represents \(N\) data sets with \(m\) features, \(\bar{x}\) and \(\sigma\) are the mean and variance of \(x\), respectively. Then, establish the correlation matrix \(R\) and calculate its eigenvalues and eigenvectors. Finally determine the number of principal components, and calculate the variance contribution rate and the cumulative variance contribution rate

$$R = X^*X^*/(N - 1)$$

(3)

$$\eta = 100\%\lambda_i/\sum_i \lambda_i$$

(4)

$$\eta_{\Sigma}(p) = \sum_i \eta_i$$

(5)

where \(X^*\) is the standardized data matrix, the eigenvalue \(\lambda_i \geq \lambda_2 \geq \cdots \geq \lambda_m\) of the autocorrelation matrix \(R\) and the corresponding eigenvector \(u_1 \geq u_2 \geq \cdots \geq u_m\) is obtained. \(\eta\) is the variance contribution rate, and \(\eta_{\Sigma}(p)\) is the cumulative variance contribution rate.

For feature selection, we mainly consider to remove the features with strong correlation because they will cause the multicollinearity of the model and make the model prediction inaccurate. The basis for feature selection mainly depends on the correlation coefficient, which is calculated as follows

$$r = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^N (y_i - \bar{y})^2}}$$

(6)

where \(x\) and \(y\) represent features, \(r\) represents the correlation coefficient between the features \(x\) and \(y\), \(\bar{x}\) and \(\bar{y}\) represent the means of \(x\) and \(y\), respectively. When the correlation coefficient \(r\) is greater than 0.9, a strong correlation is considered to exist between two features, which will lead to the multicollinearity and inaccurate prediction accuracy of the model.

3.2. Normalization Method. Different features usually have different dimensions, and the values may vary greatly. Failure to process them may affect the results of data analysis. In order to eliminate the influence of the difference between the dimensions and the value range of the indicators, it is necessary to carry out standardization processing, and the data are scaled according to the proportion to make it fall into a specific area, which is convenient for comprehensive analysis. The data normalization method used in this article is Z-score. The main purpose of Z-score is to uniformly transform data of different magnitudes into the same magnitude so that they can be uniformly measured by the calculated Z-score value to ensure the comparability of data. The mean of the processed data is 0, and the standard deviation is 1. The conversion formula is

$$X^* = \frac{X - \bar{X}}{\sigma}$$

(7)

where \(\bar{X}\) is the mean of the original data, and \(\sigma\) is the standard deviation of the original data, and \(X^*\) is the normalized value.

3.3. ML Algorithms. The following four ML methods are evaluated in this work to compare how different models predict the data:

MLR is an analysis method for modeling the relationship among one or more independent variables based on the least squares.28 To normalize the data, an optimized convex
function is constructed, and the least squares method and gradient descent method are used to calculate the final fitting parameters. Linear regression is a regression analysis that models the relationship between one or more independent variables using a least squares function. This function is a linear combination of one or more model parameters called regression coefficients. The case of only one independent variable is called simple regression, and the case of more than one independent variable is called multiple regression. It involves simply describing the relationship between output and input as a linear relationship. MLR tries to represent the output variable as a linear combination of multiple input variables

$$Y_i = \beta_0 + \sum_{j=1}^{n} \beta_j X_i$$

(8)

where $$\beta_i$$ is the linear combination coefficient, and $$X_i$$ represents the $$j$$-th characteristic of the $$i$$-th data. The training process is the process of finding the optimal values of the linear combination coefficients, which can produce the most accurate predictions corresponding to them.

We usually build an optimization function for a convex function and use the least squares and gradient descent methods to calculate the final fitting parameters. However, when the relationship between input and output is not linear, the accuracy of the MLR is limited. For the sake of simplicity, we use MLR as a baseline method for comparing other ML algorithms. Obviously, the implementation of more complex ML algorithms is meaningful only when other algorithms have more accurate predictions than MLR.

The DT is a widely used supervised learning method. It divides the data set into two or more categories sequentially by using all the values of the input variables as split conditions. The DT does not need to normalize the data when applied. The DT uses the value of the input variable $$X_i$$ as a split condition, sequentially “segmenting” the data set into two or more categories at a time. Each of the internal nodes represents a test of an attribute, with each edge representing a test result and a leaf node representing a class. The decision process starts from the root node of a DT. The data to be tested are compared with the feature nodes, and the next comparison branch is selected according to the comparison result until the leaf node is the final decision result (Figure 1).

The DT algorithm attempts to find the best split criterion for creating a tree. When solving a classification problem, all observations grouped at the same terminal node have the same output category value $$Y$$. In the case of regression, the DT algorithm divides the feature space into a number of cells, each of which has a specific output. For test data, you only need to attribute it to a unit according to the characteristics and then get the corresponding output value.

A SVM tries to find the hyperplane with the largest distance between the positive samples and the negative samples in the feature space. SVMs are also a popular ML algorithm. The basic model is defined as the linear classifier with the largest interval in the feature space. That is, the learning strategy of the SVM is to maximize the interval and finally can be transformed into a solution to a convex quadratic optimization problem. For SVM regression, the final model is determined by minimizing the deviation between the actual value and the predicted value.

The RF is composed of DT clusters. The definition of random forest is that the training subset of each tree is randomly generated, and each branch node is also randomly selected from the feature subset. For each tree, the RF method uses a self-service sampling method to conduct error estimation based on the sample data outside the bag. In order to make the base classifiers have obvious differences and mutual independence, when generating the DT, samples and features are randomly sampled to improve the diversity of the model. RF is an improvement of the DT algorithm, which consists of a collection of DTs. For quantitative prediction (regression), RF uses the population average of the values predicted by each individual DT as the predicted output value $$Y_i$$ based on the input $$X_i$$. For classification prediction, the RF prediction value $$Y$$ is the final prediction category based on the category with the highest number of occurrences per individual DT prediction. A significant advantage of RF over the DT algorithm is that it is more generalized and can avoid overfitting effectively. There are two random processes in the RF: randomly selecting data sets and randomly selecting features. The data set and feature set chosen for each tree are different, so it is convenient to explain the problem from different angles.

3.4. Division of the Database. In the process of model training, we use the cross-validation method used in the division stage of the testing set and training set to select the model to determine the best model type and model parameters. When dividing the testing set and the training set, we use random stratified sampling so that the distribution of the training set and the testing set can be kept as consistent as possible, making the model’s prediction ability as strong as possible. In addition, cross-validation can have significant effects on enhancing the generalization ability of the model. Each model is built on a different training set and validated on a corresponding test set. Based on the results of cross-validation, we finalize the selection of model types and parameters. The specific operation of cross-validation is shown in Figure 2.

3.5. Evaluation Metrics. RMSE is the root mean squared error between the predicted values and the true values, which is a quantitative trade-off. However, when the dimensions are different, it is difficult to measure the effect of the model with RMSE, which requires the use of $$R^2$$ and mean absolute error (MAE). $$R^2$$ is a measure of how well the predicted values fit the true values. $$R^2$$ is used to compare the error of model fitting with the mean of the original output data, and its range is [0, 1]. When it is 0, it means that the predicted value of the model is not as good as using the mean instead. When it is 1, it means that the model fits perfectly. The MAE is the average value of absolute error. It is actually a more general form of the
error metric, which is a measure that is less affected by outliers. The above three metrics are calculated as follows

\[
R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}
\]

(9)

\[
\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}
\]

(10)

\[
\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}
\]

(11)

where \(n\) is the total number of COFs, \(\hat{y}_i\) and \(y_i\) are the predicted value and the value computed by GCMC simulation, respectively, for the \(i\)-th COF. \(\bar{y}\) is the average value of \(y_i\).

### 3.6. Model Training

Among the above four ML algorithms, only MLR has no hyperparameters, so MLR does not require model tuning. The other three algorithms need to find suitable hyperparameters. In order to search for the best parameter configuration as quickly as possible, we set the grid search range for hyperparameter optimization of different algorithms. Due to the numerous parameters of the model, we only list some optimization ranges that have a greater impact on the model. The specific search range is shown in Table 1.

### 3.7. AutoML Methods

The current best-performing method for AutoML is based on EA and Bayesian optimization. Here, we focus on the TPOT method based on EA optimization, which can automatically explore thousands of the most cumbersome parts of ML to find the most suitable algorithm and parameters for the current data situation. In the TPOT model construction process, only the data preparation step needs to be completed manually. The subsequent feature engineering, model generation, and model evaluation are all automatically completed within the TPOT. Its overall function is shown in Figure 3. Before the model is constructed, data cleaning, feature selection, and feature construction are performed, and then, an appropriate algorithm is selected to adjust and optimize model parameters. Finally, the performance of the model is verified and evaluated through indicators such as root-mean square error and accuracy. The pretreatment mainly includes the standard scaler and robust scaler. Feature extraction uses a random principal singular value decomposition variant called randomized PCA. Feature construction is a polynomial feature construction. The feature selection contains four methods, which are the recursive feature elimination strategy, the select percentile of the top \(n\) features, the select that does not meet the minimum variance threshold, and the select \(k\)-best of the best top \(n\) features. Moreover, select \(k\)-best can be combined with chi-square tests and mutual information to select features. Their expressions are as follows

---

**Table 1. Type of ML Algorithms and Hyperparameter Values Tried and the Best Parameter**

| ML algorithm | hyperparameters | values tried | best parameter |
|--------------|-----------------|--------------|----------------|
| DT           | Max_depth       | 10, 11, ..., 20 | 18             |
| SVM          | Kernal          | rbf, linear, poly | Rbf          |
|              | C               | 0.1, 0.5, 1, 2, 3, 4, 5 | 0.5          |
| RF           | N_trees         | 100, 200, ..., 500 | 500           |
|              | Min_leaf_size   | 1, 2, 3, 4, 5, 6, 7 | 4             |

The process of optimizing model parameters uses the fivefold cross-validation method. Although the scope of parameter optimization is narrowed based on experience, the model selection and parameter optimization still require a lot of time. The main parameter that affects the performance of the DT model is the max depth of the DT. The deeper the tree, the more detailed the data set will be divided, but it will cause the risk of overfitting. Therefore, we mainly optimize the depth of the DT. The kernel function in SVM is the main factor that affects its regression performance. The kernel function is equivalent to the feature conversion function, which can map features to higher-dimensional space or infinite dimensions, making the sample easy to divide. Since the kernel function may cause dimensional disasters and overfitting, and the penalty coefficient \(C\) weighs the fitting ability and the predictive ability, the optimization of the parameters of \(C\) and the kernel function at the same time helps to find a better-performance SVM model. The main impact on the RF model is the number of DTs because the risk of RF overfitting is smaller than that of the DT model. We only tuned the minimum leaf-node parameters of the DT. Finally, according to the results of the fivefold cross-validation, the optimal parameters were selected and the final prediction model is determined in Table 1.
The models included in model selection have two directions of classification and regression, including almost all the classification and regression algorithms of scikit-learn. The description of algorithm selection and model optimization is as follows:

\[
\chi^2 = \sum_{i=1}^{k} \frac{(F_i - E_i)^2}{E_i} 
\]  

(12)

where \( p(x, y) \) is the joint distribution function of \( x \) and \( y \), and \( p(x) \) and \( p(y) \) are the marginal probability density functions of \( x \) and \( y \). \( F_i \) is the observed frequency of the \( i \)-th value of feature \( F \), and \( E_i \) is the expected frequency of the \( i \)-th value of feature \( F \).

The EA represents the ML pipeline as a tree structure, and each element represents the data processing method or the hyperparameter selected. The description of algorithm selection and model optimization is as follows:

\[
A^*, \lambda^* \in \arg\min_{A \in A, \lambda \in \Lambda} \sum_{i=1}^{k} L(A_j, D_{\text{train}}^{i-1}, D_{\text{valid}}^{i}) 
\]

(14)

where \( A = \{A^1, \ldots, A^n\} \) represents the algorithm set in the TPOT, and each element represents the data processing method or ML algorithm. \( A^j \in A (j = 1, \ldots, n) \) corresponds to the hyperparameter space \( \Lambda \), the model selection is \( k \)-fold cross-validation, and the data set is divided into \( k \) training sets \( \{D_{\text{train}}^{1}, \ldots, D_{\text{train}}^{k}\} \) and a set of \( k \) validation sets \( \{D_{\text{valid}}^{1}, \ldots, D_{\text{valid}}^{k}\} \), \( L(A_j, D_{\text{train}}^{i-1}, D_{\text{valid}}^{i}) (i = 1, \ldots, k) \) represents the error rate of the algorithm \( A^j \) with hyperparameter \( \lambda \in \Lambda \) trained on the training set \( D_{\text{train}}^{i} \) in the verification set \( D_{\text{valid}}^{i} \).

The EA is a field of natural heuristic technology, which is similar to global search. It uses a combination of mutation and crossover operators to efficiently explore the search space. Figure 3 shows the flow chart of ML optimized by the EA algorithm. EA represents the ML pipeline as a tree structure, where the root node of the tree is an estimator, and the data set or original features are used as the leaves of the tree. After the leaves, there are four types of operators: feature extraction, feature construction, feature selection, and model selection. The choice of operator is random, and the fitness is measured by RMSE and \( R^2 \). This tree-based process allows each node to randomly change the number and relationship and can realize any structure of the ML process.

In this paper, the EA follows the standard EA program. Also, the operation flow and relevant parameters are set as follows: the initial population is set to empirical value 100, and the system randomly generates a fixed number of tree pipelines at the beginning of the evolutionary process to constitute the primary population in genetic programming. Then, these pipelines are individually evaluated based on their \( R^2 \). To generate the next generation of the population, the system creates a copy of the process individual with the highest fitness level and places it in the offspring’s population until these elite individuals make up 10% of the population. The system randomly selects three individuals from the existing population and then puts them into the tournament to decide which individuals win. In this competition, the individuals with the lowest fitness are eliminated, and then, the processes with lower complexity (with less operating nodes) are selected from the remaining two individuals, and they are copied and placed in the next-generation population. Repeat this selection process until the offspring is filled. The per-individual mutation rate and crossover rate are set as 0.9 and 0.1, respectively.

When the next generation of the population is created, two pipelines are selected randomly, and the one-point crossover operator is used to copy the percentage of the pipeline. Split at a random point in the tree structure and then exchange their contents with each other. After the crossover and mutation operations are completed, the previous generation of individuals is completely deleted, and the evaluation—selection—crossover—mutation process is repeated with a fixed number of generations. The EA constantly changes the processes of finding the best, adds and changes the operation nodes, improves the adaptability, and deletes the redundant operation nodes. The single pipeline with the best performance discovered by the system during evolution will be tracked and stored in a separate location. After the operation is completed, it is used as the final optimization result of the pipeline.

3.8 Introduction to the Database. In this study, we use the COF database constructed by Lan et al. From the viewpoint of material genomics, they propose a gene partition method of genetic structural units (GSUs) with reactive sites and quasi-reactive assembly algorithms (QReaxAA) for structure generation, which uses an adaptive algorithm to determine the interlayer spacing, generating 471,990 materials containing 130 GSUs. The database contains 166,684 2D-COFs and 305,306 3D-COFs, and they are all structures without interpenetrating cross-frames. It also contains 10 unreported topologies. All structures can be obtained from the website: figshare.com/s/c7e3b7610a7b9de4210.

We use Zeo++ version 0.3 package to calculate the structural characteristics of COFs, such as the largest cavity

Figure 3. Algorithm flow of the TPOT framework covers the process of data preparation, feature engineering, model generation, and model evaluation.
diameter (LCD), density ($D_c$), accessible surface area ($S_{acc}$), and free volume ($V_{free}$). For $S_{acc}$ and $V_{free}$, the spherical radius of probe molecules is set to 1.84 Å (the kinetic radius of N$_2$) and 0.00 Å, respectively. Since the dynamic diameter of CH$_4$ is 3.8 Å, excluding the materials with PLD < 3.8 Å and VSA ≤ 0 m$^2$/g, we use the remaining 403,959 COFs.

3.9. GMC Simulation Details. The adsorption of CH$_4$ was calculated by GMC simulations at pressures of 65 bar and 5.8 bar and temperature of 298 K. Our in-house code HT-CADSS was used in all the simulations.$^{33}$ We first used $1 \times 10^7$ steps to ensure equilibrium and then perform $1 \times 10^7$ steps to sample the desired thermodynamic properties. The atoms of the COFs are described by the DREIDING force field$^{34}$ and frozen in their crystallographic positions to be considered rigid.$^{34}$ CH$_4$ was modeled as a single Lennard-Jones (LJ) interaction site, and its potential parameters are derived from the TraPPE force field. The Lorentz Berthelot mixing rule was used to calculate the LJ parameters between different atomic types, and the LJ interaction had a cutoff at 14.0 Å. Henry’s law constants and infinite dilution adsorption heat of CH$_4$ for all COFs were computed at 298 K using the Widom particle insertion method.$^{35}$ The elements will affect the interaction between adsorbent and adsorbate to a certain extent and then affect the adsorption capacity of COFs. Therefore, the element composition ratio of the frameworks used in this work has also been calculated. The parameters used in this word are listed in Table 2.

4. RESULTS AND DISCUSSION

4.1. Feature Analysis. Before using the MLR, DT, SVM, and RF algorithms, we first need to perform data preprocessing. Because there are too many features, we give priority to the processing method of feature selection and analyze the correlation between features and between features and working capacity. It is found that there is a strong correlation between many structures, as shown in Figure 4, which cause the multicollinearity problem of the model, so we choose to delete the features with high correlation between features (correlation coefficient > 0.9). One of the most relevant features in each group is retained for subsequent analysis. The correlation coefficients between different features and between features and working capacity are shown in Table 3.

According to Table 3, we rely on the related material knowledge to preserve features that have great impact on the working capacity. For the aperture feature, we delete the pore limiting diameter (PLD), global cavity diameter (GCD), and $V_c$. We preserve $\phi$ between the two features $D_c$ and $\phi$ and preserve $V_{free}$ between Vol and $V_{free}$.

After feature selection, there are 18 features left for further analysis. Among the 18 features, the top four with high correlation with working capacity are VSA, PLD, Si %, and C %. According to the scatter plot in Figure 5, the correlation between working capacity and VSA is the strongest. Also, it will become larger with the increase of VSA and the working capacity of COF. The PLD is also an important characteristic that affects the working capacity of COFs. When the working capacity is $5 \text{ Å} < \text{PLD} < 15 \text{ Å}$, it will become stronger with the increase of the PLD. When the PLD of the COF material is $10–15 \text{ Å}$, the working capacity of the COF material tends to be a high-performance value. When the value of PLD is greater than 60 Å, the working capacity no longer changes with the increase of the pore size and gradually stabilizes, which is fixed at $90 \text{ V}_{\text{STR}}/\text{V}$. In the process of material analysis, we add the analysis of the content of atomic categories. Although both Si % and C % have high correlation coefficients with working ability, the obvious relationship between them cannot be seen from the scatter plot. Therefore, we need to use ML to perform multivariate analysis on it and discover the potential relationship between various features and the working capacity of COFs. We use the feature-selected data set for modeling and analysis of ML.

Feature extraction is also a commonly used processing method in feature engineering. It is a process of dimensionality reduction through certain mapping functions. This process can not only eliminate the correlation between features but also extract high-information or nonredundant features according to certain specific indicators. Due to the high correlation between many features in the COF data, we consider using the PCA method to convert the data. The idea of PCA has two points. First, extract as much original data information as possible through linear combinations between variables, that is, use fewer principal components to replace all variables, which is dimensionality reduction. Second, the correlation between the various variables will overlap, and the principal

---

Table 2. Structural and Chemical Descriptors Used to Construct a Feature Vector for ML

| feature                        | unit       |
|-------------------------------|------------|
| largest cavity diameter      | Å          |
| (LCD)                         |            |
| pore limiting diameter        | Å          |
| (PLD)                         |            |
| global cavity diameter        | Å          |
| (GCD)                         |            |
| volumetric surface area       | m$^2$/cm$^3$|
| (VSA)                         |            |
| gravimetric surface area      | m$^2$/g    |
| (GSA)                         |            |
| unit cell-based surface area  | Å$^2$      |
| (ASA)                         |            |
| per unit cell volume          | Å$^3$      |
| density ($D_c$)               | g/cm$^3$   |
| free fraction ($\phi$)        | cm$^3$/g   |
| free volume ($V_{free}$)      | Å$^3$      |
| unit cell-based accessible    |            |
| pore volume ($V_c$)           |            |
| zero-coverage heat of adsorption ($Q_0^c$) | kj/mol |
| Henry coefficient ($k_H$)     | mol/kg/Pa  |
| hydrogen (H %)                | [number of hydrogen atoms per unit cell]/[total number of atoms] |
| carbon (C %)                  | [number of carbon atoms per unit cell]/[total number of atoms] |
| nitrogen (N %)                | [number of nitrogen atoms per unit cell]/[total number of atoms] |
| oxygen (O %)                  | [number of oxygen atoms per unit cell]/[total number of atoms] |
| fluorine (F %)                | [number of fluorine atoms per unit cell]/[total number of atoms] |
| chlorine (Cl %)               | [number of chlorine atoms per unit cell]/[total number of atoms] |
| bromine (Br %)                | [number of bromine atoms per unit cell]/[total number of atoms] |
| boron (B %)                   | [number of boron atoms per unit cell]/[total number of atoms] |
| silicon (Si %)                | [number of silicon atoms per unit cell]/[total number of atoms] |
| sulfur (S %)                  | [number of sulfur atoms per unit cell]/[total number of atoms] |
components obtained by the PCA are not correlated, which is the decorrelation. Therefore, the use of PCA can not only simplify the workload but also take into account the various indicators constructed, which is more convincing for the research results. We use SPSS statistical software to perform PCA analysis on the standardized data and obtain the variance contribution rate of each factor after feature extraction, as shown in Table 4. We regard the principal component as all the largest indexes that cover the variation information by more than 90%. We select the first 10 dimensions (91%) for further analysis. The percentage of variation information corresponding to each principal component is shown in Table 4. In addition, it can be seen in Figure 6 that the correlation coefficient between the features after PCA is significantly reduced, and the processed data achieve the purpose of eliminating the strong correlation between the features. This can not only reduce the data dimension and remove redundant information but also avoid the occurrence of multicollinearity. For systems that may bring greater computational burden, PCA provides a valuable strategy to reduce the computational cost of the model, thereby making the computational efficiency higher.

4.2. Performance of the ML and AutoML Models. We model the feature selection and feature-compressed data sets using ML algorithms: MLR, DT, SVM, and RF. According to the experiment, the model has good prediction ability when the data set reaches 20%, so we use 20% of the data set for training, and the remaining 80% is used as the testing set. In
the process of modeling using ML methods, we use Python’s scikit-learn library function. About the parameter setting, we use a small-scale grid search in the scikit-learn library function. According to two different feature processing methods, we use the above four ML algorithms to build a prediction model, and the results can be seen in Table 5. According to Table 5, it can be found that after feature processing (feature selection and PCA), the model is basically improved over the original data. PCA reduces the amount of calculation and speeds up the calculation after dimensionality reduction, but it inevitably loses some information, thereby leading to the lack of model accuracy. For different algorithms, SVM always performs the worst among four algorithms. This is because the sample size is large and the kernel function mapping dimension is very high, resulting in excessive calculation. Also, the modeling of SVM is always consuming among all algorithms, so the effect of SVM is poor. MLR does not require excessive hyperparameter adjustment, and the features used have a strong correlation with the target variable, so it can always maintain good performance. The performance of DT and RF models is close, but DT has a high risk of overfitting when modeling big data. Among these four ML algorithms, the RF always has strong fitting and generalization capabilities. Compared with several other algorithms, the effect of the TPOT is the most prominent in terms of time consumption and model accuracy. Figure 7 shows the prediction of the RF model after feature selection and PCA processing and the results of the TPOT in the raw data set.

Figure 7 shows the predicted values of the testing set under different models. The performance of the model is evaluated by comparing the predicted values and the GCMC simulation values. Figure 7a is the prediction result of RF after feature selection, and Figure 7b is the prediction result of the RF after PCA processing. Comparing the results of these two feature processes, we can see that the PCA feature extraction results are slightly inferior to the feature selection results, but the PCA uses fewer features. When dealing with massive data sets, it saves more computing time and resources. PCA processing does not rely on knowledge in the field of materials, which reduces the coupling of interdisciplinary subjects. Figure 7c represents the result of the TPOT running on the original data set, and its $R^2$, RMSE, and MAE are the best among all models. In addition, we compare the learning curve between the RF model after feature selection and the TPOT model, which can be seen in Figure 8. As the number of training examples increases, the cross-validation score of the model gradually becomes larger, and the cross-validation set score of the TPOT is always higher than that of the RF. We find that the TPOT can use less data and converge faster. The training score and cross-validation score of the model are both better than the RF. The TPOT does not require manual feature processing, model selection, and hyperparameter setting, eliminating the tedious process and obtaining the prediction model more precisely. The prediction pipeline model obtained by automatic search is superior to the manually selected ML model. It facilitates the research of nonprofessionals. We use the AutoML method to obtain models with higher prediction accuracy, which is of great significance for the screening and prediction of the working capacity of COFs.
Figure 6. Correlation thermograms of ten principal components.

Table 5. Model Comparison of Different ML Algorithms Based on Different Data Processing

|                  | MLR  | SVR  | DT   |
|------------------|------|------|------|
|                  | RMSE (V_{STP}/V) | $R^2$ | MAE (V_{STP}/V) | RMSE (V_{STP}/V) | $R^2$ | MAE (V_{STP}/V) | RMSE (V_{STP}/V) | $R^2$ | MAE (V_{STP}/V) |
| raw data         | 7.62 | 0.920 | 5.30 | 14.73 | 0.700 | 9.62 | 3.91 | 0.979 | 2.31 |
| feature selection| 7.61 | 0.921 | 5.26 | 9.03  | 0.888 | 5.47 | 3.86 | 0.979 | 2.31 |
| feature extraction| 9.86 | 0.86  | 6.54 | 13.48 | 0.749 | 8.52 | 5.23 | 0.962 | 3.01 |
| time consuming (h)| 0.6  | 78.64 | 18.26 |       |       |      |      |       |      |

|                  | MLR  | SVR  | DT   |
|------------------|------|------|------|
|                  | RMSE (V_{STP}/V) | $R^2$ | MAE (V_{STP}/V) | RMSE (V_{STP}/V) | $R^2$ | MAE (V_{STP}/V) |
| raw data         | 3.77 | 0.980 | 2.30 | 2.44  | 0.992 | 1.53 |
| feature selection| 3.47 | 0.983 | 2.23 |       |       |      |
| feature extraction| 3.71 | 0.981 | 2.30 |       |       |      |
| time consuming (h)| 21.36 |       |      |       |       | 14.45 |

Figure 7. Model prediction results of different ML methods. (a,b) are the results of the RF model after feature selection and PCA, respectively, (c) is the result of the TPOT.
AutoML is better than traditional ML. The data set results of AutoML are still the highest, and the effect of traditional ML algorithm RF is always the best, indicating that the TPOT has higher learning and prediction abilities. In addition, the TPOT eliminates the cumbersome process of model parameter tuning and combines the advantages of genetic algorithms when looking for the best model and parameter combination, which can get the best model faster, greatly improving the modeling efficiency and precision. The $R^2$ of the TPOT pipeline model is as high as 0.992, which provides a significant help for the performance prediction of the material.

Table 6 shows that under different circumstances, the prediction accuracy of the TPOT is nearly 10% higher than that of traditional ML methods. The accuracy of the TPOT model is significantly higher than the model in the original text. The TPOT has a very good effect on these public data sets and shows that AutoML has a great advantage, which provides a meaningful help for the analysis of the structure–effect relationship of materials.

5. CONCLUSIONS

In this work, we explore the effectiveness of using the TPOT tool in the AutoML technology to predict the working capabilities of COFs. Our results show that in most cases, for a set of classical learning models, the performance of hyperparameter optimization is better than the default value. Using the TPOT can automatically generate a suitable pipeline processing model and select the optimized hyperparameters, without the need for knowledge in the professional field to achieve model construction and tuning. Using MLR, DT, SVM, and RF as the control group of the TPOT, we find that the RF has the best performance among traditional ML methods, and the result of the TPOT is always better than the RF algorithm, indicating that the TPOT has higher learning and prediction abilities. In addition, the TPOT eliminates the cumbersome process of model parameter tuning and combines the advantages of genetic algorithms when looking for the best model and parameter combination, which can get the best model faster, greatly improving the modeling efficiency and precision. The $R^2$ of the TPOT pipeline model is as high as 0.992, which provides a significant help for the performance prediction of the material.

The use of AutoML in the field of materials has opened new horizon for materials scientists and improved their utilization of data from materials, thereby digging out more useful information to assist in the screening and design of materials. Overall, our work combines AutoML with material characteristic prediction to provide a lot of convenience for material researchers.

### AUTHOR INFORMATION

**Corresponding Authors**

Kunfeng Wang — College of Information Science and Technology, Beijing University of Chemical Technology,

Table 7 shows the adsorption performance of MOFs for CO$_2$ be seen from Table 7 that under different circumstances, the prediction accuracy of the TPOT is nearly 10% higher than that of traditional ML methods. The accuracy of the TPOT model is significantly higher than the model in the original text. The TPOT has a very good effect on these public data sets and shows that AutoML has a great advantage, which provides a meaningful help for the analysis of the structure–effect relationship of materials.

| Table 6. Prediction of CH$_4$ Delivery Based on Different Characteristics |
|-----------------------------|-----------------------------|-----------------------------|
| **structure features**      | **user define features**    | **structure features**      | **user define features**    |
| RMSE (cm$^3$/g) | $R^2$ | MAE (cm$^3$/g) | RMSE (cm$^3$/g) | $R^2$ | MAE (cm$^3$/g) |
| MLR                     | 25.07 | 0.931 | 21.21 | 92.10 | 0.182 | 42.47 |
| DT                      | 26.44 | 0.923 | 19.53 | 92.12 | 0.182 | 42.69 |
| SVM                     | 66.33 | 0.516 | 43.06 | 100.76 | 0.021 | 58.07 |
| RF                      | 20.25 | 0.955 | 15.69 | 71.33 | 0.510 | 41.38 |
| TPOT                    | 19.08 | 0.961 | 13.42 | 67.80 | 0.550 | 36.29 |

can be downloaded from http://pubs.acs.org/doi/suppl/10.1021/acscombsci.8b02257/suppl_file/cm8b02257_si_002.xlsx. This data set mainly includes two parts of features: one is the structural features (pore volume, maximum pore diameter, and pore limiting diameter) of MOFs, while another is user-defined features (metal type, crystal structure, metallic percentage, electronegativity ratio, and N/O). They are used to study the delivery of CH$_4$ and extract useful rules. Table 6 shows that ML methods use two types of features to predict the capacity for CH$_4$ delivery of MOFs. The algorithm results are better than the model accuracy in the original text, and the TPOT model’s results are the best. We can see that although the model effect of traditional ML algorithm RF is always the best, RF has good fitting and generalization abilities. However, the results of AutoML are still the highest, and the effect of AutoML is better than that of traditional ML. The data set used in Table 7 is about the selectivity and working capacity of MOFs to CO$_2$ in mixed gases, and the data set can be downloaded from http://pubs.acs.org/doi/suppl/10.1021/acschemmater.8b02257/suppl_file/cm8b02257_si_002.xlsx. It contains various pore chemistry and topological characteristics of MOFs to study the selectivity and working capacity to CO$_2$. Table 7 shows the adsorption performance of MOFs for CO$_2$ selectivity and working capacity in different gas mixtures. It can be seen from Table 7 that under different circumstances, the prediction accuracy of the TPOT is nearly 10% higher than that of traditional ML methods. The accuracy of the TPOT model is significantly higher than the model in the original text.

| Table 7. Prediction of MOF’s Selectivity and Working Ability of CO$_2$ in Mixed Gas |
|-------------------------------|-------------------------------|-------------------------------|
| **selectivity for CO$_2$/H$_2$** | **working capacity CO$_2$/H$_2$** | **selectivity for CO$_2$/N$_2$** |
| RMSE (mmol/kg) | $R^2$ | MAE (mmol/kg) | RMSE (mmol/kg) | $R^2$ | MAE (mmol/kg) | RMSE (mmol/kg) | $R^2$ | MAE (mmol/kg) |
| MLR               | 33.17 | 0.759 | 26.44 | 1.02 | 0.383 | 0.78 | 1.30 | 0.347 | 0.83 |
| DT                | 39.67 | 0.655 | 24.82 | 1.16 | 0.215 | 0.88 | 0.94 | 0.654 | 0.46 |
| SVM               | 59.11 | 0.235 | 43.06 | 0.60 | 0.787 | 0.37 | 1.21 | 0.431 | 0.51 |
| RF                | 39.67 | 0.655 | 24.82 | 1.16 | 0.215 | 0.88 | 0.94 | 0.654 | 0.46 |
| TPOT              | 25.40 | 0.880 | 14.21 | 0.50 | 0.871 | 0.25 | 0.59 | 0.820 | 0.29 |
Qingyuan Yang — State Key Laboratory of Organic-Inorganic Composites and Beijing Advanced Innovation Center for Soft Matter Science and Engineering, Beijing University of Chemical Technology, Beijing 100029, China; orcid.org/0000-0001-6961-2855; Email: qyyang@mail.buct.edu.cn

Huan Zhang — State Key Laboratory of Organic-Inorganic Composites, Beijing University of Chemical Technology, Beijing 100029, China

Xin Lai — College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China

Duli Yu — College of Information Science and Technology and Beijing Advanced Innovation Center for Soft Matter Science and Engineering, Beijing University of Chemical Technology, Beijing 100029, China

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.0c05990

Notes
The authors declare no competing financial interest.

ACKNOWLEDGMENTS
This work was supported in part by the National Key Research and Development Program of China under grant 2020YFC2003900, the National Natural Science Foundation of China under grant 22078004, the Big Science Project from 2020YFC2003900, the National Natural Science Foundation and Development Program of China under grant 645.

REFERENCES
(1) Zhou, H.-C.; Long, J. R.; Yaghi, O. M. Introduction to Metal-Organic Frameworks. Chem. Rev. 2012, 112, 673–674.
(2) Pardakh M.; Moharreri, E.; Wanik, D.; Suib, S. L.; Srivastava, R. Machine Learning Using Combined Structural and Chemical Descriptors for Prediction of Methane Adsorption Performance of Metal Organic Frameworks (MOFs). ACS Comb. Sci. 2017, 19, 640–645.
(3) Konstas, K.; Osl, T.; Yang, Y.; Batten, M.; Burke, N.; Hill, A. J.; Hill, M. R. Methane Storage in Metal Organic Frameworks. J. Mater. Chem. 2012, 22, 5675–5678.
(4) Menon, V. C.; Komarneni, S. Porous Adsorbents for Vehicular Natural Gas Storage: A Review. J. Porous Mater. 1998, 5, 43–58.
(5) Cote, A. P.; Benin, A. L.; Ockwig, N. W.; O'Keefe, Porous, Crystalline, Covalent Organic Frameworks. Sci 2005, 310, 1166–1170.
(6) Ding, S.; Wang, W. Covalent Organic Frameworks (COFs): From Design to Applications. Chem. Soc. Rev. 2013, 42, 548–568.
(7) Li, Y.; Yang, R. T. Hydrogen Storage in Metal-Organic and Covalent-Organic Frameworks by Spillover. AIChE J. 2010, 54, 269–279.
(8) Assfour, B.; Seifert, G. Adsorption of Hydrogen in Covalent Organic Frameworks: Comparison of Simulations and Experiments. Microporous Mesoporous Mater. 2010, 133, 59–65.
(9) Frenkel, D.; Smit, B. Understanding Molecular Simulation: From Algorithms to Applications. Phys. Today 1997, 50, 66.
(10) Guo, K.; Yang, Z.; Yu, C. H.; Buehler, M. J. Artificial Intelligence and Machine Learning in Design of Mechanical Materials; Royal Society of Chemistry, 2021.
(11) Bharati, A. K.; Ray, A.; Rai, R.; Jaiswal, A.; Khandelwal, M. Stability Evaluation of Dump Slope using Artificial Neural Network and Multiple Regression; Springer, 2021.
(12) Pradnyana, G. A; Darmawiguna, I.; Jaya, D. S.; Sasmita, A. Performance analysis of support vector machines with polynomial kernel for sentiment polarity identification: A case study in lecturer’s performance questionnaire. J. Phys.: Conf. Ser. 2021, 1810, 012033.
(13) Song, Y.-Y.; Lu, Y. Decision tree methods: applications for classification and prediction. Shanghai Arch. Psychiatry 2015, 27, 130–135.
(14) Dietterich, T. G. An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. Mach. Learn. 2000, 40, 139–157.
(15) Bender, J.; Ovtcharova, J. Prototyping Machine-Learning-Supported Lead Time Prediction Using AutoML. Procedia Comput. Sci. 2021, 180, 649–655.
(16) Quan, Y.; Mengs, W.; Hugo, J. E.; Chen, Y.; Dai, W.; Wei, T. Taking Human Out of Learning Applications: A Survey on Automated Machine Learning. 2018, arXiv:1810.13306.
(17) Olson, R. S.; Urbanowicz, R. J.; Andrews, P. C.; Lavender, N. A.; Kidd, L. C.; Moore, J. H. Automating Biomedical Data Science through Tree-Based Pipeline Optimization. 2016, arXiv:1601.07925.
(18) Liu, Y.; Zhao, T.; Ju, W.; Shi, S. Materials Discovery and Design Using Machine Learning. J. Materiomics. 2017, 3, 159–177.
(19) Fanourgakis, G. S.; Gkagkas, K.; Tylianakis, E.; Froudakis, G. E. A Universal Machine Learning Algorithm for Large-Scale Screening of Materials. J. Am. Chem. Soc. 2020, 142, 3814–3822.
(20) Kim, S.-Y.; Kim, S.-I.; Bae, Y.-S. Machine-Learning-Based Prediction of Methane Adsorption Isotherms at Varied Temperatures for Experimental Adsorbents. J. Phys. Chem. C 2020, 124, 19538–19547.
(21) Gilsuy, Z.; Sezginel, K. B.; Uzun, A.; Keskin, S.; Yıldırım, R. Analysis of CH4 Uptake over Metal–Organic Frameworks Using Data-Mining Tools. ACS Comb. Sci. 2019, 21, 257–326.
(22) Su, X.; Chen, N.; Sun, H. Automated Machine Learning Based on Radiomics Features Predicts H3 K27M Mutation in Midline Gliomas of the Brain. Neuro-Oncology 2019, 22, 393–401.
(23) Benjamin, P. E.; Bing, X. An Adaptive and Near Parameter-free Evolutionary Computation Approach towards True Automation in AutoML. 2020, arXiv:2001.10178.
(24) Le, T. T.; Fu, W.; Moore, J. H. Scaling Tree-Based Automated Machine Learning to Biomedical Big Data with a Feature Set Selector. Bioinformatics 2019, 35, 250–256.
(25) Tsmardinos, I.; Fanourgakis, G. S.; Greasidou, E.; Klontzas, E.; Froudakis, G. E. J. M.; Materials, M. An Automated Machine Learning Architecture for the Accelerated Prediction of Metal-Organic Frameworks Performance in Energy and Environmental Applications. Microporous Mesoporous Mater. 2020, 306, 160–173.
(26) Chandrashekar, G.; Sahin, F. A Survey on Feature Selection Methods. Comput. Electr. Eng. 2014, 41, 16–28.
(27) Liu, K. Measurement Error and Its Impact on Partial Correlation and Multiple Linear Regression Analyses. Am. J. Epidemiol. 1988, 127, 864–874.
(28) Holzinger, A. Data Mining with Decision Trees: Theory and Applications. Online Inf. Rev. 2015, 39, 437–438.
(29) Algamal, Z. Y.; Qasim, M. K.; Lee, M. H.; Ali, H. J. C.; Systems, I. L. Improving grasshopper optimization algorithm for hyper-parameters estimation and feature selection in support vector regression. Chemometr. Intell. Lab. Syst. 2021, 208, 104916.
(30) Zhou, Z.; Zhang, Y. Applications, Flood disaster risk assessment based on random forest. algorithm 2021, 6. DOI: 10.1007/s00521-021-00757-6.
(31) Lan, Y.; Han, X.; Huang, H.; Tong, M.; Huang, H.; Yang, Q. Materials Genomics Methods for High-Throughput Construction of COFs and Targeted Synthesis. Nat. Commun. 2018, 9, 5274–5283.
(32) Willems, T. F.; Rycroft, C. H.; Kazi, M. Algorithms and Tools for High-Throughput Geometry-Based Analysis of Crystalline Porous Materials. Microporous Mesoporous Mater. 2012, 149, 134–141.
(33) Mayo, S. L.; Olafson, B. D.; Goddard, W. A. DREIDING—a
generic force-field for molecular simulations. *J. Phys. Chem.* 1990,
94, 8897–8909.

(34) Fischer, M.; Hoffmann, F.; Froba, M. New Microporous
Materials for Acetylene Storage and C₂H₂/CO₂ Separation: Insights
from Molecular Simulations. *ChemPhysChem* 2010, 11, 2220–2229.

(35) Vlugt, T. J. H.; García-Pérez, E.; Dubbeldam, D.; Ban, S.;
Calero, S. Computing the Heat of Adsorption using Molecular
Simulations: The Effect of Strong Coulombic Interactions. *J. Chem.
Theory Comput.* 2008, 4, 1107–1118.

(36) Martinsson, P. G.; Rokhlin, V.; Tygert, M. A Randomized
Algorithm for the Decomposition of Matrices. *Appl. Comput. Harmon.
Anal.* 2011, 30, 47–68.

(37) Pedregosa, F.; Varoquaux, G.; Gramfort, A. Scikit-Learn:
Machine Learning in Python. *J. Mach. Learn. Res.* 2011, 12, 2825–
2830.

(38) Ryther, A.; Jacob, R.; Edwin, A.; Achay, B. Role of Pore
Chemistry and Topology in the CO₂ Capture Capabilities of MOFs:
From Molecular Simulation to Machine Learning. *Chem. Mater.* 2018,
30, 6325–6337.