MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **93** (2025) 5–40
ISSN: 0340–6253
doi: 10.46793/match.93-1.005W

OS-MSWGBM: Intelligent Analysis of Organic Synthesis Based on Multiscale Subtraction Weighted Network and LightGBM

Lanfeng Wang[a,*], Yanhui Guo[b,*], Zelin Zhang[c], Meng'en Qin[b], Zixin Li[b], Xiaoli Sun[d], Xiaohui Yang[b,†]

[a] *School of Mathematics and Statistics, Anyang Normal University, Anyang, Henan, 455000*

[b] *Henan Engineering Research Center for Artificial Intelligence Theory and Algorithms, School of Mathematics and Statistics, Henan University, Kaifeng, China, 475000*

[c] *School of Computer and Information Engineering, Henan University, Kaifeng, China, 475000*

[d] *School of Mathematical Sciences, Shenzhen University, Shenzhen, China, 518000*

xhyanghenu@163.com

(Received March 11, 2024)

Abstract

Organic synthesis plays a vital role in optimizing existing drugs and innovating new drugs. As a significant and challenging research frontier in the field of organic synthesis, cross-coupling reactions have also attracted considerable attention. In the past few years, machine learning has realized great potential in predicting the performance of cross-coupling reactions. However, most of the existing machine learning predictions are based on the two-dimensional feature information of the cross-coupling reactions. In order to obtain

[*] Lanfeng Wang and Yanhui Guo contributed equally to this work.
[†] Corresponding author.

the coupling reaction feature in a multifaceted way, we exploit the three-dimensional features of the molecules based on the molecular stick-and-ball model and the persistent homology analysis of topological data, respectively. On this basis, a weighted light convolutional neural network with multi-scale subtraction (OS-MSW) is proposed to extract the deep abstract features of the input data, and the extracted abstract features are applied to LightGBM for yield prediction, thus constructing a highly efficient prediction system OS-MSWGBM. In addition, the interpretability of the OS-MSW model is analyzed in this paper. The experiments demonstrate that OS-MSWGBM exhibits higher efficiency and more accurate prediction results, as well as notably stable prediction performance, which can provide reliable decision-making information for experimental personnel or organizations.

1 Introduction

As an essential and challenging research in the field of organic synthesis, the cross-coupling reaction has also attracted considerable attention, and its products are widely used in chemical biology, materials science and the pharmaceutical industry. In the past few decades, the rapid development of transition metal-catalyzed cross-coupling reactions, particularly those catalyzed by palladium (Pd), has been a significant advancement. This type of reaction is highly efficient, exhibits good selectivity, and operates under mild reaction conditions, making it an effective tool in modern organic synthesis.

The Buchwald-Hartwig amination reaction is one of the hotspots in the field of palladium (Pd)-catalyzed cross-coupling reactions [1–4]. However, the routes of such reactions are usually complicated, and the traditional experiments rely on a lot of manual trials and modification, which is time-consuming and costly, and the toxic by-products generated by the reaction can cause serious environmental pollution. Machine Learning (ML) has a unique advantage in reducing costs and increasing efficiency, and its performance is centered on the ability to represent data and the interpretability of results. The importance of machine learning and its huge superiority over traditional statistical methods in terms of higher accuracy and elimination of the need for a large number of hazardous experiments have caught more attention from researchers in the field of chemistry and chemical engineering. Machine learning has shown great potential for applications in drug discovery [5], molecular material design [6], reaction prediction [7], inverse synthesis design [8], and automated synthesis [9], etc. Using machine learning tools, the prediction of catalyst activation performance [10–12], chemical reaction performance [13, 14], and compound properties [15, 16] has been realized. Research in chemistry is also shifting to a data-based scientific discovery paradigm, in which researchers use computer to convert chemical data into descriptors. In this way, research in the field of chemistry can be transformed into data-based research, hence reducing the reliance on human resources to a certain extent.

In 2018, D. T. Ahneman et al. [7] transformed molecular structures into descriptors that could be recognized and calculated by computers and obtained reaction yield data under different reaction conditions through a high-throughput experimental platform. They then used random forests to predict the yield of Buchwald-Hartwig cross-coupling reactions, achieving a prediction accuracy of =7.8. This represents advanced research in the field of multidimensional chemical space prediction using machine learning methods, opening up a new path for research in the chemical field. In 2021, L. C. Peng et al. [17] supplemented and improved from the perspectives of machine learning and statistics, using quantile regression forest probability density prediction models to predict the yield of Buchwald-Hartwig cross-coupling reactions, attaining prediction intervals for the yield at different quantiles, extending the point predictions of D. T. Ahneman et al. [7] to interval predictions. In 2022, J. Dong et al. [18] proposed a feature selection method based on importance and correlation, successfully reducing feature dimensions, and employed the XGBoost (eXtreme Gradient Boosting) model to predict reaction yields. In the same year, X. C. Mu et al. [19] combined the high-dimensional characteristics of chemical reaction data and proposed a deep forest-based prediction method for cross-coupling reaction yields. Deep learning, as a dominant force in machine learning, has shown tremendous potential over the past few decades, and efforts have been made to use deep learning to solve chemical-related problems. In 2021, Y. N. Zhao et al. [25] from Dalian University of Technology used the dataset reported by D. T. Ahneman et al. [7] to predict the performance of Buchwald-Hartwig cross-coupling reactions using Deep Convolutional Neural Networks (DCNN). In 2022, H. X. Hou et al. [25] proposed the AM-1D-CNN model (Attention Mechanism 1D convolutional neural network) to predict reaction yields, further improving prediction accuracy.

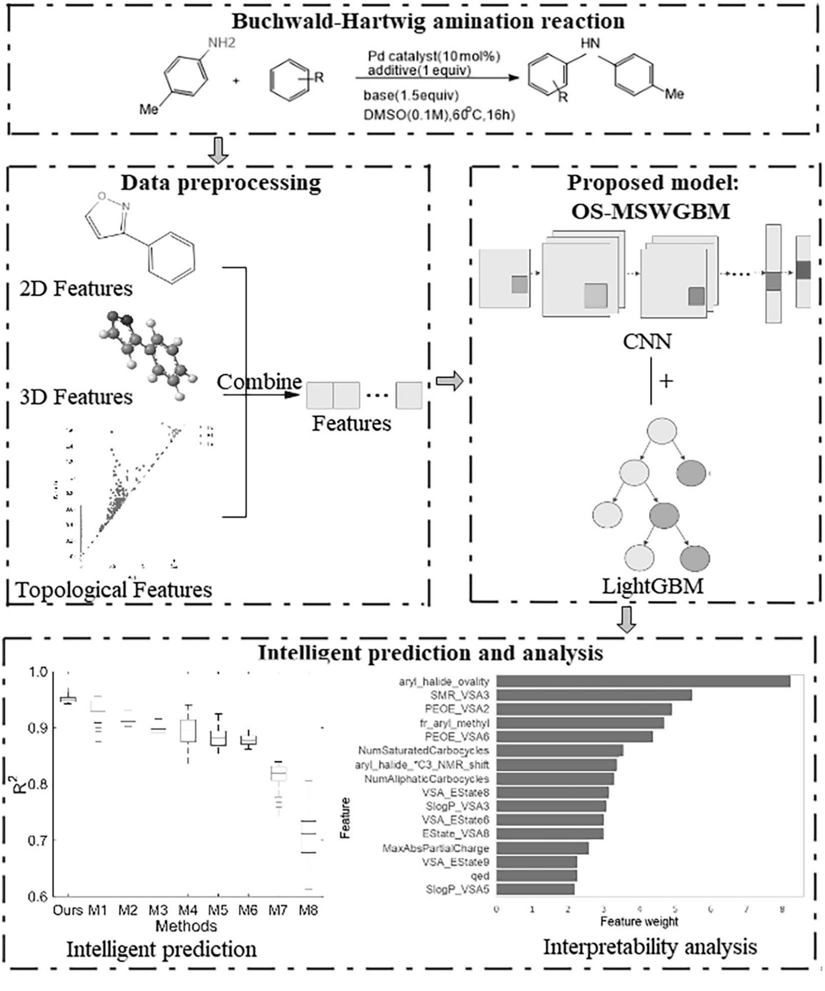


Figure 1. Flow chart of intelligent prediction and analysis.

But the relevant work of L. C. Peng et al. [17], Y. N. Zhao et al. [25], and H. X. Hou et al. [26] all input the two-dimensional feature descriptors generated by the D. T. Ahneman team, without fully considering other features of the molecules, such as three-dimensional and topological features. While the random forest algorithm used by D. T. Ahneman et al. achieved intelligent prediction of yield and interpretability to some extent, the algorithm is outdated and slow in training. The XGBoost used by J. Dong et al. [18] incurs a relatively large time cost due to the pre-sorted algorithm. The network structure designed by Y. N. Zhao et al. [26] is complex and lacks deep feature digging.

Building on the progress of these works, the focus of our study is to extract the three-dimensional(3D) and topological features of molecules, improve intelligent yield prediction models, and analyze the interpretability of the models. The main contributions are as follows: (1) Extract the three-dimensional and topological features of cross-coupling reactions based on molecular stick-ball models and persistent homology of topological data. (2) Design a lightweight deep learning model, OS-MSW (Multi-scale subtraction, multi-scale weighted and CNN for organic synthesis), based on multi-scale subtraction weighted network, and fuse it with the LightGBM model to construct an efficient yield intelligent prediction analysis system, OS-MSWGBM (Organic Synthesis based on Multi-scale Subtraction Weighted, CNN and LightGBM). (3) Conduct feature attribution to analyze the interpretability of the OS-MSW(Organic Synthesis based on Multi-scale Subtraction Weighted and CNN) model.

2 Intelligent analysis and prediction model for reaction yield—OS-MSWGBM

To achieve efficient yield prediction, this paper proposed an intelligent organic synthesis system called OS-MSWGBM. Firstly, a lightweight deep learning model OS-MSW, based on CNN with multi-scale subtraction weighted, is designed. The abstract features extracted by the OS-MSW feature learning network are then input into the LightGBM model for training and prediction. Finally, feature attribution is conducted to provide interpretability analysis of the OS-MSW model. OS-MSWGBM is an innovative approach for research and synthesis processes in the field of organic chemistry, offering reliable decision-making information for experimental researchers or institutions.

This section will introduce the construction of three-dimensional features, topological features, and OS-MSWGBM model as well as three evaluation metrics.

2.1 Feature construction

Existing methods for predicting the yield of Buchwald-Hartwig cross-coupling reactions still depend on two-dimensional features and have not fully considered other feature information of the coupling reaction. For the purpose of obtaining the feature information of the Buchwald-Hartwig cross-coupling reaction from multiple perspectives, we will extract the three-dimensional features and topological features of the coupling reaction based on the molecular stick-and-ball model and persistent homology of topological data. By cascading two-dimensional features, three-dimensional features, and topological features, we can describe compounds from multiple angles, thereby improving the accuracy of prediction.

2.1.1 Construction of three-dimensional features

Two-dimensional features mainly quantify the physical properties of compounds, like the highest (lowest) occupied molecular orbital energy level, molecular dipole moment, electronegativity, hardness, atomic charges, nuclear magnetic resonance shifts, vibrational frequencies, and intensities. Differently, three-dimensional features quantify the spatial structural and physicochemical properties of compounds, such as molecular weight, valence electron count, molecular fingerprint density, octanol-water partition coefficient, molecular ring compactness index, geometric complexity and diversity of the molecule, polarity, and solubility. The construction steps of three-dimensional features are as follows:

1) Arrange all variables involved in the Buchwald-Hartwig amination reaction, including halides, ligands, bases, and additives, in a certain order

and draw the two-dimensional structure diagrams of each combination.

2) Use Chem3D software to convert the drawn two-dimensional structure diagrams of each combination into three-dimensional structure diagrams (hockey stick structure) and save them as mol format files.

3) Use the RDKit toolkit to calculate and output the three-dimensional structure descriptors of each mol format file in Python, thereby quantifying the structural features and physicochemical properties of organic compounds.

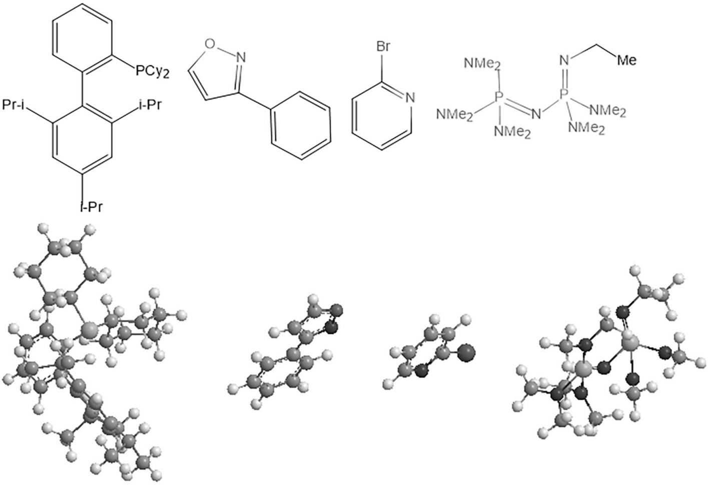


Figure 2. Two-dimensional structures of the four reactants and corresponding stick-ball structures.

2.1.2 Construction of topological features based on persistent homology

The extraction of topological features based on persistent homology involves identifying and analyzing the topological invariants such as average lifetimes and persistent entropy of connected components, cycle structures, and void structures.

As the parameter ε increases from 0, these topological features, such as connected components (connected branches), cycle structures (1-dimensional loops), and void structures (2-dimensional voids), are tracked from birth to extinction. The longer these topological features persist, the more useful they are for data analysis. If a topological feature emerges and disappears quickly, it is more likely to be noise. This process is known as persistent homology. The fundamental concepts of persistent homology, include simplex, simplicial complex, homology, and filtration.

(1) Simplices and simplicial complexes

Simplices are the basic geometric structures in topological data analysis [27], extending the concept of triangles. They are used to approximate complex shapes and are mathematically and computationally easier to handle than primitive shapes.

Simplicial complexes describe the topological structure of a set of points. There exist various definitions for different data types and different types of complex structures have different properties. Most commonly used type is the Vietoris-Rips complex (VR complex) [27], since the VR complex is easily extendable to higher dimensions and provides a simple and intuitive representation of the building process, making it suitable for computational analysis. Therefore, VR complexes are used throughout the data analysis in this paper.

The construction of a VR complex:

(a)Building a neighbor graph. The neighbor graph is an undirected weighted graph, denoted by (G, ω), where $G = (V, E)$ indicates undirected graph, V is the set of vertices, E is the set of edges, and ω denotes the weight. Let $E_\varepsilon = \{\{u, v\} \,|d\,(u, v) \leq \varepsilon, u \neq v \in V\}$, where $d\,(u, v)$ is the distance function between two points. Furthermore, the map $\omega : E \to R$ makes each edge weight the distance between the two points, i.e., $\omega\,(\{u, v\}) = d\,(u, v)\,, \forall\,\{u, v\} \in E_\varepsilon(V)$.

(b)Expanding the VR complex in the neighbor graph obtained in step (a). Given a neighbor graph, construct the VR complex $(R(G), \omega)$ using the weight filtration ω, where:

$$R\,(G) = V \cup E \cup \left\{ \sigma | \binom{\sigma}{2} \subseteq E \right\}. \tag{1}$$

In summary, the VR complex is the union of all vertices, edges, and simplex σ in the neighbor graph, where any combination of two points in the vertices of simplex σ belongs to the set of edges E.

(2) Homology and filtration

To construct and utilize these simplicial complexes in data analysis, it is necessary to further compute meaningful topological features within these complexes. Topological Data Analysis (TDA) employs tools from persistent homology theory to compute the number of connected components and n-dimensional cycles (such as holes in circles or voids in spheres) in the topological space established from the dataset. This requires the computation of homology groups and Betti numbers denoted as β [28].

For a simplicial complex K, a k chain can be represented in a summation form of K: $\sum\limits_{i=1}^{N} c_i \left[\sigma_i^k\right]$, where σ_i^k is a k-dimensional simplex in the simplicial complex K, $c_i \in Z_2$. The collection of all k chains in K forms an abelian group $C_k(K)$.

We now can extend the definition of the boundary operator introduced in Equation 2 to chains. The boundary operator applied to a k-chain c_k is defined as

$$\partial_k c_k = \sum a_i \partial_k \sigma_i, \tag{2}$$

the boundary operator is a map from c_k to c_{k-1}, which is also named boundary map for chains. Note that operator ∂_k satisfies the property that $\partial_k \circ \partial_{k+1}\sigma = 0$ for any $(k+1)$-simplex σ following the fact that any $(k-1)$-face of σ is contained in exactly 2 k-faces of σ. The chain complex is defined as a sequence of chains connected by boundary maps with an order of decaying in dimensions and is represented as

$$\cdots \to C_n(K) \overset{\partial_{n-1}}{\to} \cdots \overset{\partial_1}{\to} C_0(K) \overset{\partial_0}{\to} 0. \tag{3}$$

The k-cycle group and k-boundary group are defined as kernel and image of ∂_k and ∂_{k+1}, respectively,

$$\begin{aligned} B_k &= \operatorname{Im} \partial_{k+1} = \left\{ \partial_{k+1} c | c \in C_{k+1} \right\}, \\ Z_k &= Ker \partial_{k+1} = \left\{ \partial_{k+1} c | c \in C_{k+1} \right\}, \end{aligned} \tag{4}$$

where Z_k is the k-cycle group and B_k is the k-boundary group. Since $\partial_k \circ$

$\partial_{k+1} = 0$, we have $B_k \subseteq Z_k \subseteq C_k$. With the aforementioned definitions, the k-homology group is defined to be the quotient group by taking k-cycle group modulo of k-boundary group as

$$H_k = Z_k / B_k, \tag{5}$$

where H_k is the k-homology group. The kth Betti number is defined to be rank of the k-homology group as

$$\beta_k = rank(H_k) = rank(Z_K) - rank(B_k). \tag{6}$$

The Betti number β_k exists because $\beta_k \leq rank(Z_p) < \infty$. Betti numbers are important invariants of a topological space. β_0 represents the number of connected components and β_1 represents the maximum number of cuts along closed curves that can be made while keeping the space connected.

The Betti numbers obtained through homology groups can effectively describe the topological structure of a simplicial complex. Intuitively, the homology group H_0 represents connected components, H_1 represents cyclic structures, and H_2 represents void structures. The k-th Betti number β_k denotes the number of k-dimensional voids.

The filtration of a simplicial complex K refers to a nested sequence of subcomplexes of K, reflecting the data structure at different scales:

$$\varnothing = K_0 \subseteq K_1 \subseteq \cdots \subseteq K_m = K. \tag{7}$$

Through the filtration process of a simplicial complex, the homology groups of each simplicial complex can be gained, and topological features can be computed using these homology groups. During the filtration, topological features that persist for a long time are more likely to be important attributes of the object we study. In other words, non-boundary cycles that do not quickly map to the edge are probably vital features of the internal structure of the data, which is the persistence of topological features.

The persistence diagram [29] can provide a visualization of persistent homology analysis, illustrating the birth and death of topological features.

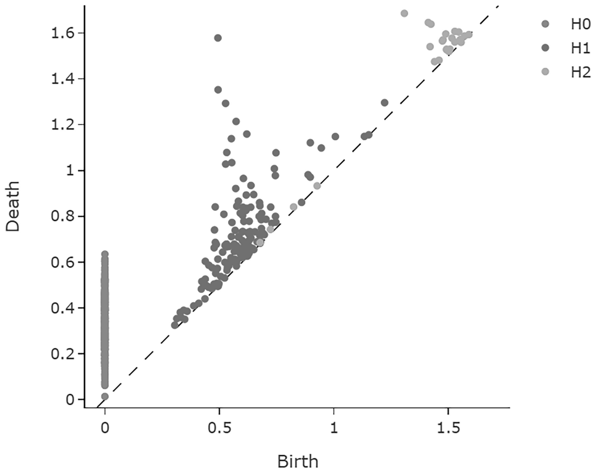


Figure 3. Persistence diagram.

(3) Construction of Topological Feature

In topological machine learning, topological methods concentrate on invariants. Topological invariants are topological structures where properties stay unchanged under a specific type of continuous transformation, e.g., connected components, loop structures, and void structures. The barcodes or persistence diagrams of topological invariants are transformed into structured features represented as quantized vectors, which are then input into machine learning algorithms to train models. The persistence diagram represents the results of persistent homology analysis as pairs of birth and death times, with the horizontal and vertical axes representing the birth and death values of the topological invariants, respectively. If we use a_j^k, b_j^k to symbolize the birth and death of the j-th topological invariant of the k-dimensional homology group, then l_j^k represents its lifetime; $l_j^k = b_j^k - a_j^k$ where $k \in \mathbb{N}, j = 1, 2, \cdots, N_k$ is the total number of topological invariants in the k-dimensional homology group. Table 1 summarizes some methods for constructing topological features.

Table 1. Construction of topological feature.

Feature name	Dimension of homology group	Feature description
Num_rel_holes	0,1,2	The number of components, cycle structures, and void structures with a survival time greater than 50% of their respective maximum survival times.
Num_holes	0,1,2	The total number of existing components, cycle structures, and void structures.
Sum_lengh	0	The sum of survival times for all components.
Avg_lifetime	0,1,2	The average survival period of components, cycle structures, and void structures.
Length_betti	0	The survival period of the second longest-lasting component.
Onset_longest_betti	1	The birth time of the longest-lasting cycle structure.
Polynomial_feature_1	0	$\frac{1}{N}\sum_{i=1}^{N} b_i(d_i - b_i)$.
Polynomial_feature_2	0	$\frac{1}{N}\sum_{i=1}^{N} b_i^2(d_i - b_i)^4$.
Persistence entropy	0,1,2	$E(D) = -\sum\limits_{\alpha \in A} p_\alpha \log p_\alpha$, $p_\alpha = \frac{(d_\alpha - b_\alpha)}{\sum\limits_{\alpha \in A}(d_\alpha - b_\alpha)}$.
Persistence landscape	0,1,2	The p-Wasserstein distance between two persistence diagrams D_1 and D_2 is the infimum over all bijections $\gamma : D_1 \cup \Delta \to D_2 \cup \Delta$ of $\left(\sum\limits_{x \in D_1 \cup \Delta} \lVert x - \gamma(x) \rVert_\infty^p\right)^{1/p}$, where $\lVert - \rVert_\infty$ is defined for $(x, y) \in R^2$ by $\max\{\lvert x \rvert, \lvert y \rvert\}$.
Wasserstein distance	0,1,2	The p-Wasserstein distance between two persistence diagrams.
Bottleneck distance	0,1,2	The limit $p \to \infty$ defines the bottleneck distance. More explicitly, it is the infimum over the same set of bijections of the value $\sup\limits_{x \in D_1 \cup \Delta} \lVert x - \gamma(x) \rVert_\infty$.

2.2 OS-MSWGBM

This section presents a light, multi-scale subtraction network with a CNN backbone. To extract crucial features without significantly increasing model complexity, multi-scale weighted is incorporated, therefore, the multi-scale subtraction weighted network, OS-MSW, is constructed. The final fully connected layer is replaced with LightGBM in order to mitigate overfitting risks associated with fully connected layers and improve model interpretability and prediction performance. The resulting hybrid model OS-MSWGBM, enhances prediction efficiency and significantly reduces runtime while intensifying deep feature exploration.

2.2.1 Multi-scale subtraction network

Scale features greatly contribute to capturing the contextual information of objects. Inspired by scale space theory, an increasing number of multi-scale methods have been proposed to address natural scale variations. Current approaches gradually fuse different scale features through addition or concatenation, which could lead to substantial redundancy and weaken the complementarity among features at different scales. Aiming at resolving the generation of redundant information, we apply a subtraction unit when fusing features at different levels.

Suppose X_A and X_B represent adjacent level feature maps. They all have been activated by the ReLU operation. We define a basic subtraction unit (SU):

$$SU = Conv(|X_A \ominus X_B|), \tag{8}$$

where $\ominus$ is the element-wise subtraction operation, $|\cdot|$ calculates the absolute value and $Conv(\cdot)$ denotes the convolution layer. The SU unit can capture the complementary information of X_A and X_B and highlight their differences, thereby providing richer information for the decoder.

To obtain higher-order complementary information across multiple feature levels, we horizontally and vertically concatenate multiple SUs to calculate a series of differential features with different orders and receptive fields. The detail of the multi-scale subtraction module can be found in

Figure 4. We aggregate the level-specific feature (M_1^i) and cross-level differential features ($M_{n\neq 1}^i$) between the corresponding level and any other levels to generate complementarity enhanced feature (C_i). This process can be formulated as follows:

$$C_i = Conv(\sum_{n=1}^{5-i} M_n^i), i = 1, 2, 3, 4. \tag{9}$$

Finally, all C_i participate in decoding.

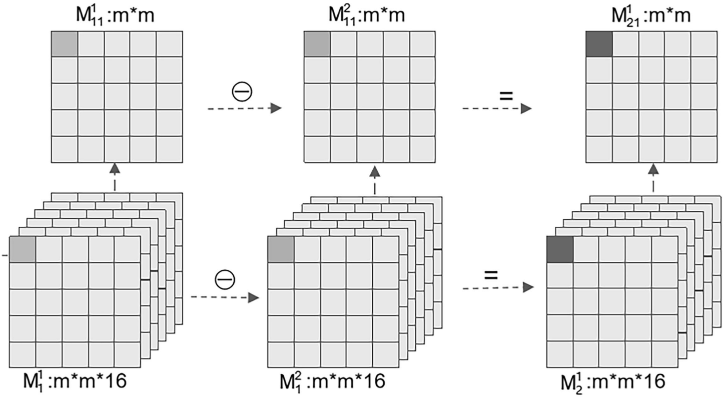


Figure 4. SU subtraction.

2.2.2 Multi-scale weighted

Supposing there are i feature layers X_i, each with k feature maps X_{ik}. First, perform global average pooling on X_{ik} to obtain its average value:

$$X_{ik\_mean} = \frac{1}{|R|} \sum_{(p,q)\in R} X_{ik(p,q)}, \tag{10}$$

where $X_{ik(p,q)}$ represents the element located at (p, q) in region R of the k-th feature map of the i-th layer, $|R|$ represents the total number of elements in the k-th feature map. Thus the average value $X_{i\_mean} = (X_{i1\_mean}, X_{i2\_mean}, \cdots, X_{ik\_mean})$ of each feature layer, is processed by a fully connected layer using the sigmoid activation function to obtain the corresponding weight:

$$X_{i\_weight} = sigmoid(W_i * X_{i\_mean} + b_i), \tag{11}$$

where W_i and b_i represent the weight and bias of the i-th fully connected layer. Reshape X to a shape of (1,1,k), and then weight the i feature layers by multiplying them with the corresponding weights to get the weighted feature layers $X_{i\_weighted}$:

$$X_{i\_weighted} = X_i * X_{i\_weight}, \tag{12}$$

Thus, the final weighted average value can be:

$$X_{weighted} = X_{1\_weighted} + X_{2\_weighted} + \cdots + X_{i\_weighted}, \tag{13}$$

which can be used as the model's input for the next classification and regression tasks.

2.2.3 OS-MSWGBM prediction model

Based on what is discussed above, a carefully-designed feature learning network is constructed. Firstly, we propose a multi-scale subtraction network with a two-dimensional CNN as the backbone. And then without significantly increasing the complexity of the model, multi-scale weighting is added, making the network structure more scientifically efficient. This feature learning network possesses powerful feature learning capabilities of convolutional neural networks, promotes the complementary abilities among features of different scales, as well as the ability to focus on core features through multi-scale weighting.

However, it should be noted that the OS-MSW model ultimately uses traditional fully connected layers, which require a huge number of neurons and parameters. This results in two major limitations: on the one hand, the model requires amounts of parameters, leading to computational complexity; on the other hand, an excessive number of neurons in the fully connected layers also easily increases the risk of overfitting. The advantage of the LightGBM model lies in its ease of parameter tuning, resistance to overfitting, and fast model training speed. Therefore, to reduce the risk of overfitting and the associated complex computations in the fully connected layers, the OS-MSW model is combined with the LightGBM model. Specifically, the abstract features extracted by the OS-MSW function as input for training and prediction in the LightGBM model, while the rest of the structure maintains unchanged. The final objective function is

$$L(\varphi) = \sum l(\hat{y}_i, y_{_i}), \tag{14}$$

whose idea is to iteratively generate multiple weak models and then add the predictions of each weak model together, with each subsequent model $f_t(x)$ being generated from the previous learning model $f_{t-1}(x)$.

Since $\hat{y}_i = \phi(x_i) = \sum\limits_{k=1}^{K} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i)$, the objective function can be transformed into:

$$L^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)). \tag{15}$$

In summary, the workflow of OS-MSWGBM is as follows:

(1) Data preprocessing. Standardize the data and reshape the n-dimensional dataset into an $m * m$ descriptor matrix, where $m * m = n$. If the dimension n is not enough to be converted into a matrix $m * m$, it can be padded with zeros.

(2) Model training. Set the loss function of the OS-MSWGBM feature learning network as Mean Square Error (MSE), and use the Adam optimization algorithm. Continuously optimize loss through the optimization algorithm to decrease the error value, update the parameters of the neural network through error backpropagation, and save the network parameters when convergence is achieved. Then, use the abstract features extracted from the third fully connected layer as training data to input into the LightGBM model for training.

(3) Model testing. The well-trained OS-MSW is used to extract features from the test set. Subsequently, the extracted abstract features are input into LightGBM to calculate the predicted yield $\hat{y}$. Regression fitting analysis is then performed using evaluation metrics $R^2(y, \hat{y})$, $RMSE(y, \hat{y})$, $MAE(y, \hat{y})$ and actual yield y to evaluate the model.

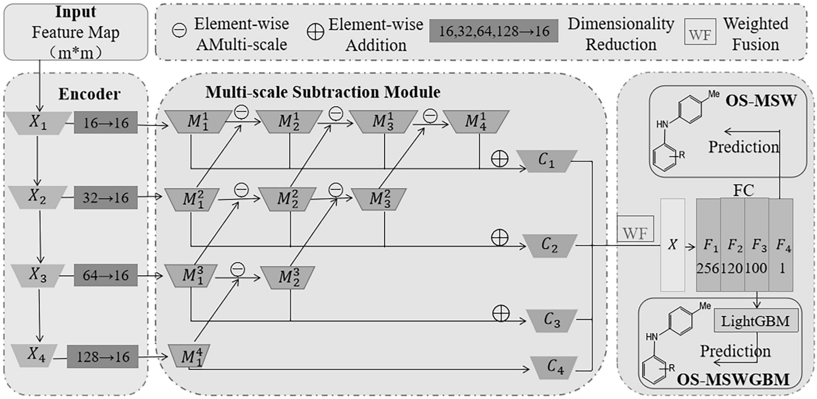


Figure 5. workflow of OS-MSWGBM hybrid model.

2.2.4 Feature attribution

By using deep models, we attain higher performance through greater abstraction (more layers) and tighter integration (end-to-end training). Nevertheless, structural complexity makes these models difficult to interpret. Therefore, deep models are striking a balance between interpretability and accuracy. In this research, we utilize the Gradient Weighting technique to multiply the gradient of the input with respect to the feature block by the weight of the feature block itself, attaining attribution weights for each input. These attribution weights indicate the contribution of eigenvalues in the input eigenmatrix to the feature block, and can thus be used to output the weight values of each feature descriptor. Calculation steps are:

Assume there is a deep neural network model, where $f(x)$ represents the output of the input x. We hope to trace the contribution of the input x to the model output $f(x)$ by Gradient Weighting technique.

First, calculate the gradient of the model output $f(x)$ with respect to the input x, i.e., $\nabla_x f(x)$.

Next, compute the contribution of each input feature to the model output. To do this, multiply the gradient $\nabla_x f(x)$ with the feature block A to obtain:

$$\alpha_i = \sum_{j,k} \nabla_{A_{j,k}} f(x) * A_{i,j,k}, \tag{16}$$

where $A_{i,j,k}$ denotes the weight of the i-th channel, j-th row, and k-th column in the feature block A.

Then, normalize α_i to obtain the attribution weights:

$$w_i = \frac{\alpha_i}{\sum\limits_i \alpha_i}. \tag{17}$$

Finally, multiplying the attribution weights w_i by the input feature x_i, we get the contribution of each feature

$$L_{grad-CAM}(x)_c = \sum_i w_i * x_i. \tag{18}$$

2.3 Evaluating indicators

In the regression prediction of yield, R^2, Root Mean Square Error ($RMSE$) and Mean Absolute Error (MAE) are selected to measure the regression prediction effect of the model.

(1) R^2, also known as coefficient of determination, reflects the interpretable proportion of the independent variable to the dependent variable. The value range of R^2 is between 0 and 1. The closer R^2 is to 1, the better the fitting effect of the model.

$$R^2 = 1 - \frac{SSR}{SST} = 1 - \frac{\sum\limits_i (y_i - \hat{y}_i)^2}{\sum\limits_i (y_i - \bar{y})^2}, \tag{19}$$

where SST is the sum of squares, and the sum of squares of errors between the original data y_i and the mean value $\bar{y}$ is calculated. SSR is the sum of squares of regression, which calculates the sum of squares of the mean value $\bar{y}$ and the error of fitting data $\hat{y}_i$.

(2) $RMSE$ is the square root of the ratio of the square of the deviation between the observed value $\hat{y}_i$ and the real value y_i and the observation times n. The smaller the value of $RMSE$, the better the regression pre-

diction effect of the mode.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{20}$$

(3) MAE is the average of the absolute value of the error between the observed value and the real value. Similarly, it is used to measure the deviation between the predicted value and the real value. The smaller the MAE value, the better the regression prediction effect of the model.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{21}$$

3 Experiments

In this section, three-dimensional features and topological features of the Buchwald-Hartwig coupling reaction were extracted, and cascaded them with 2D features selected based on feature importance and correlation. The cascaded feature descriptor data is applied to test and analyze the convergence and prediction of the OS-MSWGBM model.

3.1 Data sources

This paper selects the data with regard to the Buchwald-Hartwig cross-coupling reaction published by D. T. Ahneman et al. [7]. The authors generated a total of 3960 sets of valid experimental data through high-throughput experiments (with 5 sets of experimental data missing yield values, which are excluded from this study). To avoid time-consuming analysis and recording of computational data, D. T. Ahneman et al. [7] developed the software Spartan to calculate molecular, atomic, and vibrational property, and then extracted these features from the resulting text files for general user access. Spartan extracted a total of 120 feature descriptors to symbolize each reaction, which can be further categorized into three types: molecular descriptors (28), atomic descriptors (64), and vibrational descriptors (28).

However, many descriptor may exhibit significant correlations, leading to overfitting and an increase in computational time. The feature descriptors selected by J. Dong et al. [18] based on feature importance and correlation as input data for all subsequent algorithms were used in this paper. The study has already suggested that the attained 21 descriptors can effectively replace the original 120 descriptors. Hence, unless otherwise specified, the 2D features used in this paper are all based on the selected 21 descriptors.

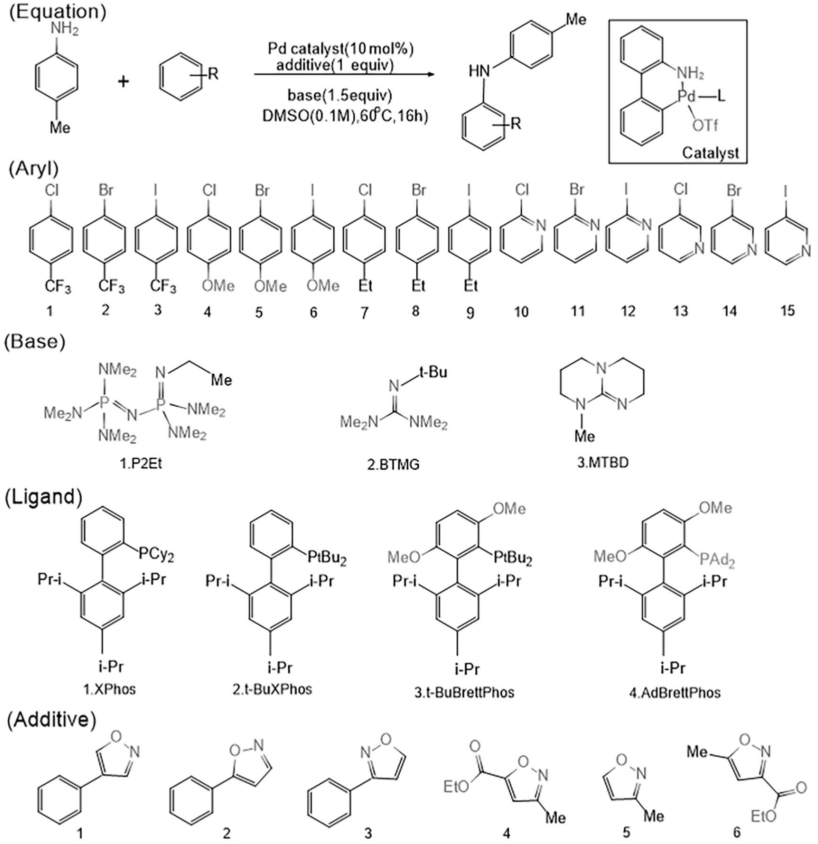


Figure 6. Reaction components of the Buchwald-Hartwig amination reaction for the first 821 samples.

It is an extremely time-consuming and resource-intensive task generating plenty of samples (3960). In practical scenarios, researchers may sometimes only have access to relatively few samples. We conducted experiments using a small subset of samples from the data published by D. T. Ahneman et al. [7] on the Buchwald-Hartwig cross-coupling reaction, specifically 821 samples. Such sample size is sufficient to support our research while reducing resource consumption during the process.

Experimental environment: each experiment is the result of an average of 100 trials with the same configuration, training set: testing set = 7:3. Computer configuration is as follows: Brand: Dell; CPU: Intel(R) Core (TM) i7-7700HQ CPU @2.80GHz(8CPUs), 2.8GHz; Memory type: DDR4.Software: under Python3.7 scikit-learn module or MATLAB R2020a on a 2.80GHz machine with 24.00GB RAM.

3.2 Feature analysis

To obtain feature descriptors for the Buchwald-Hartwig cross-coupling reaction from multiple perspectives, we conduct the extraction of three-dimensional (3D) and topological features. Specifically, 208 3D features are extracted from molecular stick-and-ball models of the cross-coupling reaction. 37 topological features are extracted by leveraging persistent homology analysis based on the 3D features. Due to zeros in some of the 3D descriptors, 10 columns were removed. Subsequently, the 21 two-dimensional (2D) feature descriptors selected by J. Dong et al. [18] are concatenated with the remaining 3D and topological features. It has been previously demonstrated that these 21 2D feature descriptors efficiently replace the original 120 descriptors [17, 18]. Eventually, this paper have a total of 256 feature descriptors.

Furthermore, as shown in Figure 7, a comparison of the CNN prediction results for the 2D features, 3D features, the concatenation of 3D and topological features (3D+T), and the concatenation of 2D, 3D, and topological features (2D+3D+T) reveals that the concatenation of 2D, 3D, and topological features (2D+3D+T) achieved superior prediction accuracy. This finding underscores the effectiveness of the extracted 3D and topological features. Therefore, the concatenated data (256 feature descriptors) will be used as the experimental data for the subsequent sections of this study.

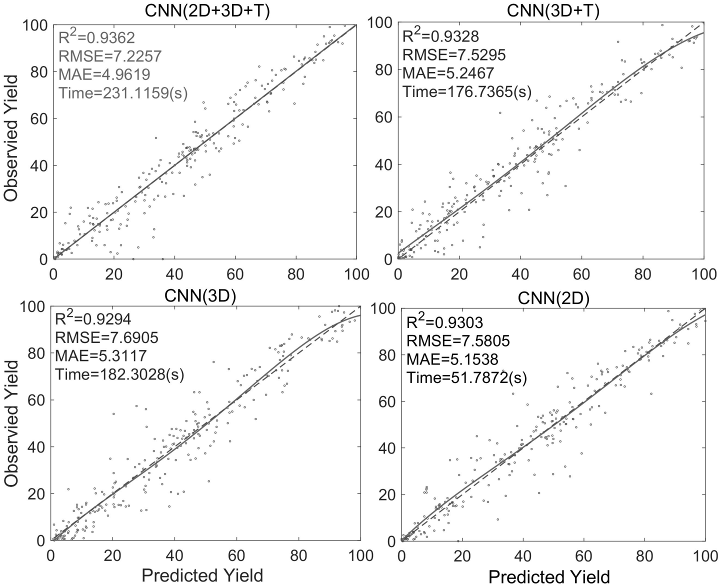


Figure 7. Comparison of CNN prediction results with different feature descriptors.

3.3 Performance of OS-MSWGBM-based chemical reaction yield prediction

This section will investigate the convergence and predictive performance of the OS-MSWGBM. Through comparisons with other state-of-the-art models, we have demonstrated that OS-MSWGBM outperforms most other models in predictive accuracy as well as operation speed. Moreover, the model's generalization performance has been validated through out-of-fold predictions and Suzuki-Miyaura dataset predictions.

3.3.1 Convergence analysis

Figure 8 visualizes the changes in the average root mean square error and average root mean square absolute error of the training and testing sets over the iterations in ten-fold cross-validation. It is evident that the error curves for both the training and testing sets show a decreasing trend with increasing iteration numbers, ultimately stabilizing. This indicates that the OS-MSWGBM model converges after training.

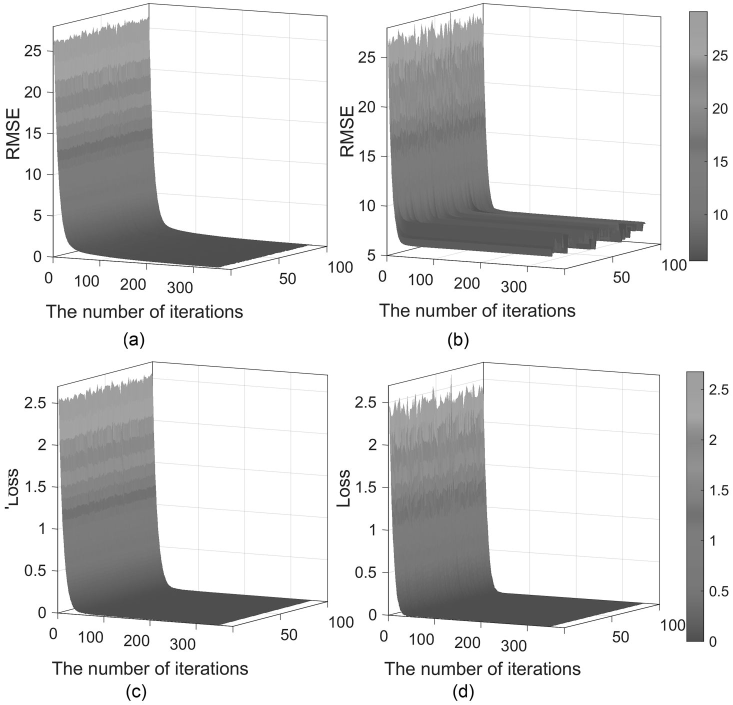


Figure 8. (a)The $RMSE$ of the training set varies with the number of iterations. (b)The $RMSE$ of the testing set varies with the number of iterations. (c)The absolute error between adjacent iteration steps of the training set. (d)The absolute error between adjacent iteration steps of the testing set.

3.3.2 Model performance analysis

To better fit real data, this section employ ten-fold cross-validation and grid search methods to obtain the optimal parameters for the LightGBM model. Additionally, this study compares the results from eight other regression methods, including CNN, XGBoost, Random Forest, Extra tree, AdaBoost, Gradient Boost, Support Vector Machine Regression (SVR), and Multilayer Perceptron Regression (MLPR). From Figure 9, it is obvious that the OS-MSWGBM model demonstrates superior predictive capabilities, requiring only 0.9418 seconds runtime. In contrast, the conventional CNN model runs 245 times slower than OS-MSWGBM, representing a 99.59% improvement in runtime speed. This comes down to the fact that CNN models typically utilize traditional fully connected layers, which demand numbers of neurons and parameters. On the other hand, the LightGBM model leverages Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) techniques, allowing for accelerated training without compromising accuracy.

While OS-MSWGBM model needs relatively longer runtime compared to XGBoost, Gradient Boost, MLPR, and SVR models, it still outperforms them in terms of predictive accuracy. Specifically, it shows improvements of 4.05%, 8.21%, 16.43%, and 34.49% in prediction accuracy, with corresponding reductions in $RMSE$ of 24.26%, 35.40%, 47.26%, and 58.52%, as well as reductions in MAE of 22.07%, 38.71%, 50.69%, and 61.11%, respectively. Overall, OS-MSWGBM displays nearly perfect performance, combining higher operational efficiency with more accurate predictive outcomes.

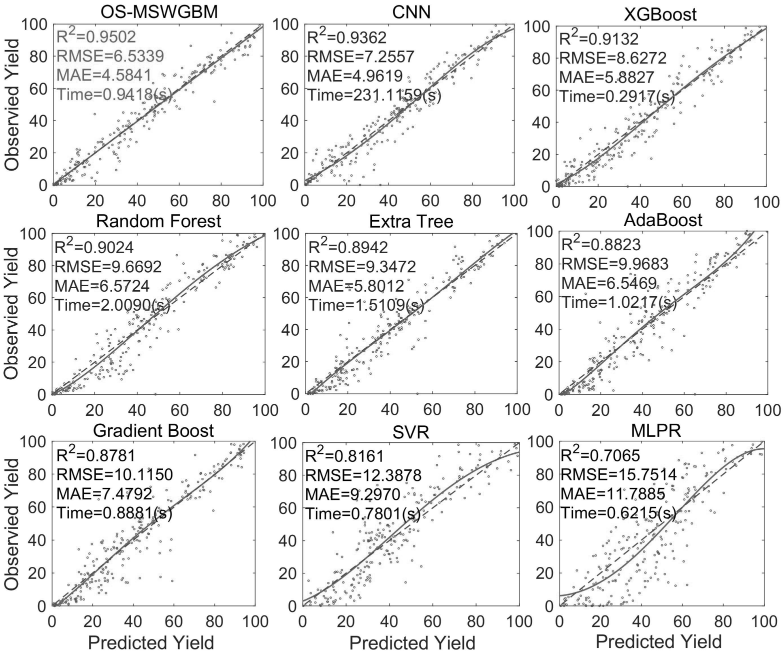


Figure 9. Prediction results of different models.

The boxplot of results from 100 experiments, as shown in Figure 10, indicates that OS-MSWGBM outstrips the other eight methods (M1: CNN, M2: XGBoost, M3: Random Forest, M4: Extra tree, M5: AdaBoost, M6: Gradient Boost, M7: SVR, and M8: MLPR) in terms of both smaller R^2 and larger $RMSE$ and MAE. In addition, the boxplot for the 100 experiment results is narrower, meaning more concentrated predictive outcomes and more stable predictive performance.

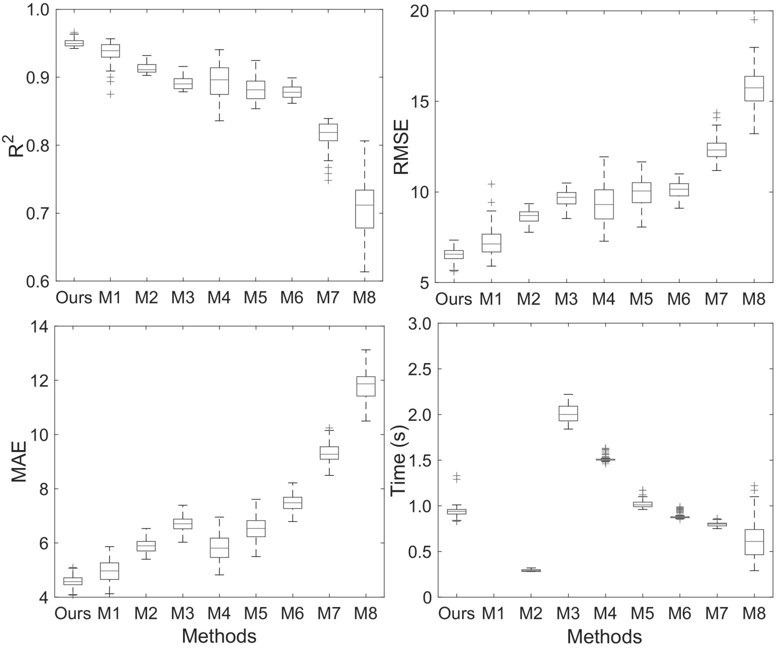


Figure 10. Box plots of 100 prediction results for different models.

3.4 Ablation experiments

The ablation experiments are conducted to further validate the superior performance of OS-MSWGBM. As illustrated in Figure 11, the predictive accuracy of OS-MSWGBM surpassed that of other models. Although the inclusion of LightGBM do not cause a significant improvement in predictive accuracy, it notably shortens runtime, achieving a speed increase of nearly 80 times. Additionally, it becomes apparent that each module we introduce (multi-scale subtraction MS, multi-scale weighting MW, LightGBM) is effective.

Table 2. Ablation experiments.

Symbol	Model	CNN	MS	MW	LightGBM
Ours	OS-MSWGBM	✓	✓	✓	✓
A1	OS-MSW	✓	✓	✓	
A2	OS-MSGBM	✓	✓		✓
A3	OS-MS	✓			✓
A4	OS-MWGBM	✓		✓	✓
A5	OS-MW	✓		✓	
A6	OS-GBM	✓			✓
A7	CNN	✓			
A8	LightGBM				✓

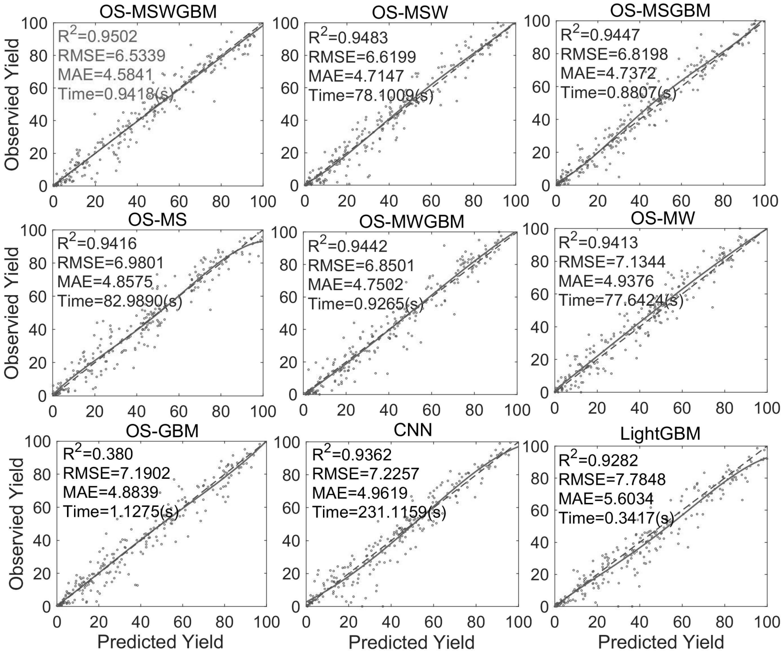


Figure 11. Prediction results of ablation experiments.

The boxplots in Figure 12 display the results from 100 ablation experiments. Similarly, when comparing OS-MSWGBM with the eight methods (A1: OS-MSW, A2: OS-MSGBM, A3: OS-MS, A4: OS-MWGBM, A5: OS-MW, A6: OS-GBM, A7: CNN, and A8: LightGBM), OS-MSWGBM exhibits both larger R^2 and smaller $RMSE$ and MAE. The boxes representing the results of the 100 ablation experiments for OS-MSWGBM are smaller, indicating more concentrated predictive outcomes. This indirectly proves the effectiveness of each module we introduce (multi-scale subtraction MS, multi-scale weighting MW, LightGBM).

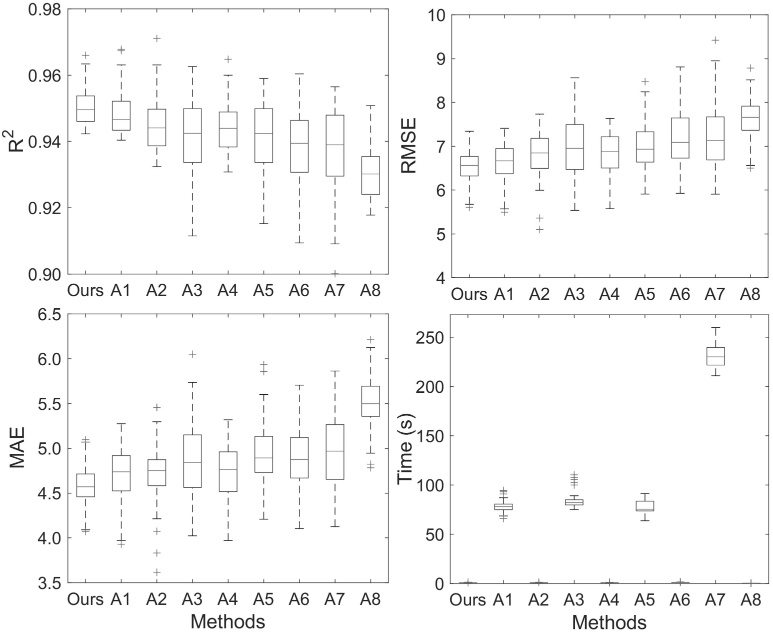


Figure 12. Boxplot of 100 prediction results for ablation experiments.

3.5 Generalization analysis

To validate the generalization performance of the proposed method, two experiments are conducted: (1) conducting out-of-fold predictions on the same dataset and (2) testing predictions on Suzuki-Miyaura dataset

3.5.1 Out-of-fold predictions

Out-of-sample predictions refer to making predictions on samples that are not used during model training, and then estimating the model's performance in predicting new data. This approach is useful for predicting with new data (data not seen during training) and assessing model performance. It allows for the evaluation of the model's generalization performance. Scoring the model's predictions made during each training iteration and then calculating the average score is the most common method for model evaluation.

In addition to averaging the prediction evaluations for each model, out-of-fold predictions also aggregate the predictions for each model into a list, which includes a summary of the reserved data used as the test set for each training group. After all model training is completed, this list is used as a whole to obtain a single accuracy score. This method is used considering that each data point appears only once in each test set. Each sample in the training dataset has a prediction during the cross-validation process. Therefore, all predictions can be collected, compared with the target results, and scores can be calculated at the end of the entire training. This approach highlights the model's generalization performance more effectively.

Table 3. Comparison of out-of-fold prediction results of OS-MSWGBM, CNN, and LightGBM.

Methods	R^2	$RMSE$	MAE	$Time(s)$
OS-MSWGBM	0.9588	5.8249	4.1238	1.1226
CNN	0.9451	6.4562	4.5157	196.9825
LightGBM	0.9397	7.0575	5.0407	0.4108

From Table 3, it can be observed that compared to CNN and LightGBM, OS-MSWGBM exhibits larger R^2 and smaller $RMSE$ and MAE, with much shorter running time. Figure 13 shows the fitting and error plots of the predicted values versus the original values for OS-MSWGBM, CNN, and LightGBM. It is apparent from the plots that OS-MSWGBM has the best fitting effect, with overall small error.

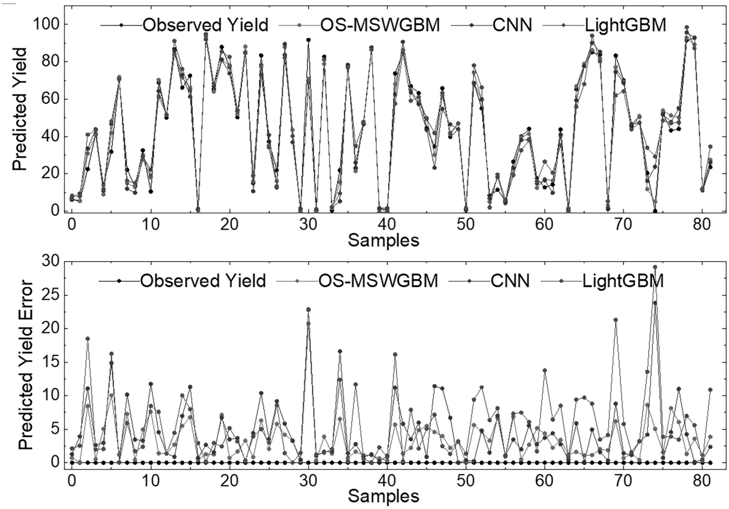


Figure 13. Comparison of out-of-fold results of OS-MSWGBM, CNN, LightGBM.

3.5.2 Suzuki-Miyaura dataset predictions

To confirm the generalization performance of our method, test is conducted on the Suzuki-Miyaura dataset, which originates from an article by the Pfizer team [30]. The authors conducted high-throughput screening of Suzuki-Miyaura C-C coupling reactions, comprising 11 reactants, 12 ligands, 8 bases, and 4 solvents, resulting in 5760 reactions. The predictive target was the reaction yield, but Pfizer and others did not apply a machine learning model in the original study. In 2018, Cornin et al. [31] reported machine learning exploration of this dataset, using one-hot encoding for reaction encoding and training a two-layer neural network to predict reaction yields. In 2021, Gong et al. [32] used GAT for direct image input for feature learning and prediction. In this paper, according to encoding of Cornin et al. we use the OS-MSWGBM model for prediction.

From Table 4, we can see that OS-MSWGBM shows a decrease in accuracy compared to the DeepReac model, with a decrease of 1.40% in R^2 and an increase of 4.55% in $RMSE$. Nonetheless, running speed of DeepReac is over 1000 times that of OS-MSWGBM, indicating a 99.91% improvement in speed for OS-MSWGBM. The lower accuracy of CNOS-

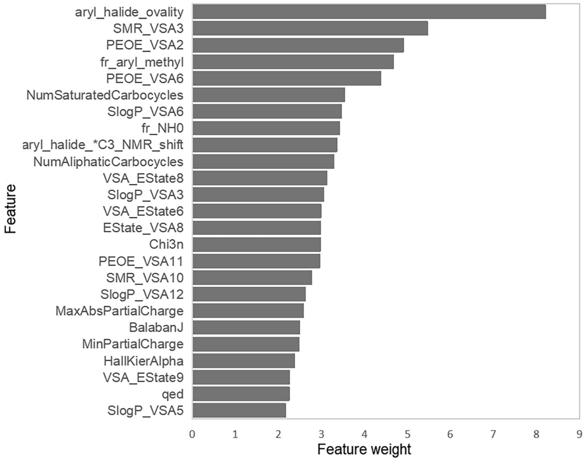
MSWGBM compared to DeepReac may be due to the one-hot encoding data used by CNOS-MSWGBM not fully describing molecular features, whereas DeepReac uses GAT to directly prossess molecular features. Undoubtedly, we are willing to compromise a small amount of accuracy to improve model efficiency.

Table 4. Comparison of test results for Suzuki-Miyaura dataset

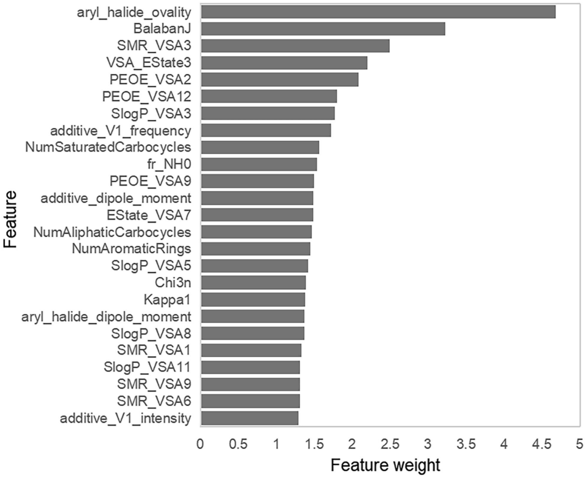
Methods	R^2	$RMSE$	MAE	$Times(s)$
OS-MSWGBM	0.8884	0.092	0.0641	1.69s
Neural network(NN) [31]	0.8413	0.1115	0.0755	18.09s
DeepReac [32]	0.901	0.088	$--$	About 3h

3.6 Interpretability analysis

Figure 14 is the feature weight plots for two samples in the test set, displaying the top 25 ranked feature weights for each sample. Notably, features such as aryl_halide_ovality and SMR_VSA3 play crucial roles in predicting outcomes in the OS-MSW. Analysis of the feature weights for each test set sample finds that aryl_halide_ovality and SMR_VSA3 etc., consistently rank among the top features, indicating their significant impact within the OS-MSW. More exactly, the prediction error for the first sample is only 0.0861, while the error for the second sample is considerably higher at 4.8495. This disparity may be attributed to the fact that for the second sample, important features such as fr_aryl_methyl, PEOE_VSA6, and SlogP_VSA6 have weights ranked outside the top 50, suggesting that the significance of these features is overlooked for the second sample, thus contributing to the increased prediction error. Also, this observation indirectly underscores the effectiveness of the extracted geometric features (e.g., SMR_VSA3, PEOE_VSA2).



(a)



(b)

Figure 14. Feature weighting diagram.

3.7 Software of OS-MSWGBM

As shown in Figure 15, for the convenience of users, we has developed a EXE software called OS-MSWGBM, which can implement 3D features, topological features, intelligent prediction of reaction yields and interpretability analysis. Specifically as follows: (1)Read the compound's .mol format file, click on [3D Features] for three-dimensional feature extraction. (2)Read the compound's three-dimensional feature data, click on [Topological Features] for extracting topological features. (3)Click on [Train Model] to start training the OS-MSWGBM model, the training progress will be displayed in [progress], click on [Stop Training] at any time to terminate the training. (4)Click on [Intelligent Prediction] for intelligent prediction with the OS-MSWGBM model. (5)Finally, click on [Interpretability Analysis] to trace features and conduct interpretability analysis of the OS-MSW model. Users can selectively click on the desired results. The tool is powerful, easy to use, and offers very high accuracy in results, providing convenience for chemists researching cross-coupling reactions.

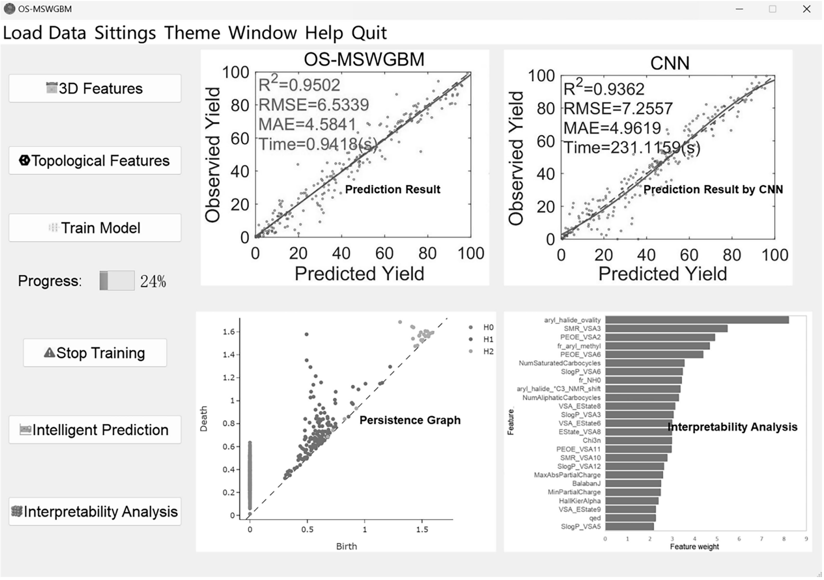


Figure 15. The system interface of OS-MSWGBM.

4 Conclusion

This paper presents the construction of a more efficient organic chemical synthesis system OS-MSWGBM, based on convolutional neural networks and LightGBM. First, molecular stickball models and persistent homology from topological data analysis are used to extract three-dimensional and topological features of molecules, which are then concatenated with original two-dimensional features to obtain comprehensive feature information for the Buchwald-Hartwig cross-coupling reaction. Subsequently, a multi-scale subtraction network with a two-dimensional CNN as its backbone is designed, incorporating multi-scale weighting to focus on key features without significantly increasing model complexity, therefore, the multi-scale subtraction weighted network, OS-MSW, is constructed. The learned features are then fed into LightGBM to construct the efficient prediction model, OS-MSWGBM. OS-MSWGBM enhances feature expression through feature re-representation, while the introduction of LightGBM accelerates the model's operational efficiency. Experimental results demonstrate that the OS-MSWGBM prediction model not only exhibits higher operational efficiency and more accurate prediction results in forecasting reaction yields, but also displays more stable predictive performance. In addition, this paper conducted feature tracing analysis based on OS-MSW to identify input features remarkably influencing reaction yields, aiding advancements in the field of chemistry and providing more accurate assistance to experimenters.

Similarly, the proposed intelligent predictive analysis system can be applied to other chemical reactions beyond the Buchwald-Hartwig cross-coupling reaction. In the future, combining the feature interpretability of LightGBM with the feature tracing of OS-MSW appears another intriguing and challenging avenue for researchers.

References

[1] P. Ruiz-Castillo, S. L. Buchwald, Applications of palladium-catalyzed C-N cross-coupling reactions, *Chem. Rev.* **116** (2016) 12564–12649.

[2] J. F. Hartwig, Evolution of a fourth generation catalyst for the amination and thioetherification of aryl halides, *Acc. Chem. Res.* **41** (2018) 1534–1544.

[3] D. S. Surry, S. L. Buchwald, Biaryl phosphane ligands in palladium-catalyzed amination, *Angew. Chem. Int. Ed.* **47** (2008) 6338–6361.

[4] M. M. Heravi, Z. Kheilkordi, V. Zadsirjan, M. Heydari, M. Malmir, Buchwald-Hartwig reaction: an overview, *J. Org. Chem.* **861** (2018) 17–104.

[5] J. M. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, Automatic chemical design using a data-driven continuous representation of molecules, *ACS Cent. Sci.* **4** (2018) 268–276.

[6] M. Fujinami, J. Seino, T. Nukazawa, S. Ishida, T. Iwamoto, Virtual reaction condition optimization based on machine learning for a small number of experiments in high-dimensional continuous and discrete variables, *Chem. Lett.* **48** (2019) 961–964.

[7] D. T. Ahneman, J. G. Estrada, S. Lin, S. D. Dreher, A. G. Doyle, Predicting reaction performance in C-N cross-coupling using machine learning, *Science* **360** (2018) 604–610.

[8] M. H. S. Segler, M. Preuss, M. P. Waller, Planning chemical syntheses with deep neural networks and symbolic AI, *Nature* **555** (2018) 604–610.

[9] J. M. Granda, L. Donina, V. Dragone, D. L. Long, L. Cronin, Controlling an organic synthesis robot with machine learning to search for new reactivity, *Nature* **559** (2018) 377–381.

[10] Z. Ahmadvand, M. Bayat, M. A. Zolfigol, Toward prediction of the precatalyst activation mechanism through the cross-coupling reactions: Reduction of Pd (II) to Pd (0) in precatalyst of the type PdPEPPSI, *J. Comput. Chem.* **41** (2020) 2296–2309.

[11] W. Yang, T. T. Fidelis, W. H. Sun, Prediction of catalytic activities of bis(imino) pyridine metal complexes by machine learning, *J. Comput. Chem.* **41** (2020) 1064–1067.

[12] S. Kite, T. Hattori, Y. Murakami, Estimation of catalytic performance by neural network — product distribution in oxidative dehydrogenation of ethylbenzene, *Appl. Catal. A* **114** (1994) L173–L178.

[13] S. Ishioka, I. Miyazato, L. Takahashi, T. N. Nguyen, T. Taniike, High-throughput experimentation and catalyst informatics for oxidative coupling of methane, *ACS Catal.* **10** (2022) 921–932.

[14] A. Mayr, G. Klambauer, T. Unterthiner, S. Hochreiter, DeepTox: Toxicity prediction using deep learning, *Front. Environ. Sci.* **3** (2016) 1–15.

[15] S. Ryu, Y. Kwon, W. Y. Kim, A Bayesian graph convolutional network for reliable prediction of molecular properties with uncertainty quantification, *Chem. Sci.* **36** (2019) 5438–5446.

[16] M. Druchok, D. Yarish, S. Garkot, Ensembling machine learning models to boost molecular affinity prediction, *Comput. Biol. Chem.* **93** (2021) #107529.

[17] L. C. Peng, J. Dong, X. C. Mu, Z. L. Zhang, Y. Q. Zhang, X. H. Yang, Intelligent predicting reaction performance in multidimensional chemical space using quantile regression forest, *MATCH Commun. Math. Comput. Chem.* **87** (2022) 299–318.

[18] J. Dong, L. C. Peng, X. H. Yang, Z. L. Zhang, P. Y. Zhang, XGBoost based intelligence yield prediction and reaction factors analysis of amination reaction, *J. Comput. Chem.* **43** (2022) 289–302.

[19] X. C. Mu, J. Dong, L. C. Peng, X. H. Yang, Deep forest-based intelligent yield predicting of Buchwald-Hartwig coupling reaction, *MATCH Commun. Math. Comput. Chem.* **88** (2022) 5–27.

[20] I. Arel, D. Rose, T. Karnowski, Deep machine learning – a new frontier in artificial intelligence research, *IEEE Comput. Intell. Mag.* **5** (2010) 13–18.

[21] G. B. Goh, N. O. Hodas, A. Vishnu, Deep learning for computational chemistry, *J. Comput. Chem.* **38** (2017) 1291–1307.

[22] A. Mater, M. Coote, Deep learning in chemistry, *J. Chem. Inf. Model.* **59** (2019) 2545–2559.

[23] M. Afzal, A. Sonpal, M. Haghighatlari, A. Schultz, J. Hachmann, A deep neural network model for packing density predictions and its application in the study of 1.5 million organic molecules, *Chem. Sci.* **10** (2019) 8374–8383.

[24] B. Liu, B. Ramsundar, P. Kawthekar, J. Shi, J. Gomes, Q. Luu, S. Ho, J. Sloane, P. Wender, V. Pande, Retrosynthetic reaction prediction using neural sequence-to-sequence models, *ACS Cent. Sci.* **3** (2017) 1103–1113.

[25] Y. N. Zhao, X. C. Liu, H. Lu, X. F. Zhu, T. H. Wang, G. Luo, R. C. Zheng, Y. Luo, An optimized deep convolutional neural network for yield prediction of Buchwald-Hartwig amination, *Chem. Phys.* **550** (2021) #111296.

[26] H. X. Hou, H. Z. Wang, Y. H. Guo,P. Y. Zhang, L. C. Peng, X. H. Yang, Regression prediction of coupling reaction yield based on attention–driven convolutional neural network, *MATCH Commun. Math. Comput. Chem.* **89** (2023) 199–222.

[27] C. S. Pun, K. Xia, S. X. Lee, Persistent-homology-based machine learning and its applications – A survey, *arXiv* (2018) 1–42.

[28] A. Zomorodian, G. Carlsson, Computing persistent homology, *Discr. Comput. Geom.* **33** (2005) 249–274.

[29] P. Bendich, J. S. Marron, E. Miller, A. Pieloch, S. Skwerer, Persistent homology analysis of brain artery trees, *Ann. Appl. Stat.* **10** (2016) #198.

[30] D. Perera, J. W. Tucker, S. Brahmbhatt, C. J. Helal, A. Chong, W. Farrell, P. Richardson, N. W. Sach, A platform for automated nanomole-scale reaction screening and micromole-scale synthesis in flow, *Science* **359** (2018) 429–434.

[31] J. M. Granda, L. Donina, V. Dragone, D. L. Long, L. Cronin, A. Zomorodian, Controlling an organic synthesis robot with machine learning to search for new reactivity, *Nature* **559** (2018) 377–381.

[32] Y. Gong, D. Xue, G. Chuai, J. Yu, Q. Liu, DeepReac+: deep active learning for quantitative modeling of organic chemical reactions, *Chem Sci.* **12** (2021) 14459–14472.