Scilight

Smart Chemical Engineering

https://www.sciltp.com/journals/sce

Article

AI-Accelerated Catalyst Selection and Operating Conditions Optimization for Hydrocracking

Siying Liu [1,†], Zheyuan Pang [2,†], Cheng Lian [1,2,*], Chong Peng [3,*], Xiangchen Fang [4] and Honglai Liu [1,2]

[1] State Key Laboratory of Chemical Engineering and Low-Carbon Technology, School of Chemistry and Molecular Engineering, East China University of Science and Technology, Shanghai 200237, China
[2] State Key Laboratory of Chemical Engineering, Shanghai Engineering Research Center of Hierarchical Nanomaterials, School of Chemical Engineering, East China University of Science and Technology, Shanghai 200237, China
[3] State Key Laboratory of Fine Chemicals, School of Chemical Engineering, Dalian University of Technology, Dalian 116024, China
[4] Dalian Research Institute of Petroleum and Petrochemicals, SINOPEC, Dalian 116024, China
* Correspondence: liancheng@ecust.edu.cn (C.L.); pengchong@dlut.edu.cn (C.P.)
† These authors contributed equally to this work.

How To Cite: Liu, S.; Pang, Z.; Lian, C.; et al. AI-Accelerated Catalyst Selection and Operating Conditions Optimization for Hydrocracking. *Smart Chemical Engineering* **2025**, *1*(1), 4. https://doi.org/10.53941/sce.2025.100004

Received: 4 August 2025
Revised: 29 August 2025
Accepted: 28 September 2025
Published: 24 October 2025

Abstract: Hydrocracking is a critical refining technology for upgrading heavy oils, where catalyst selection and operating condition adjustment are crucial for enhancing catalytic performance and product quality. Currently, this matching process relies heavily on the experimental method, which is time-consuming and resource-intensive. Data-driven methods provide a solution for this problem. However, the application of data-driven methods demands specialized data science expertise. This work utilized GPT-4 as an AI assistant to facilitate the development and interpretation of data-driven models for hydrocracking catalysis, establishing the relationship between catalyst properties, feedstock characteristics, operating conditions, and hydrocracking tail oil properties. Gradient-weighted class activation mapping was employed to identify key factors influencing the properties of tail oil. Based on the model's prediction, the impacts of replacing catalysts and adjusting operating conditions on tail oil properties were explored. The framework in this study is expected to reduce experimental iterations by 60%, highlighting the potential of AI in optimizing hydrocracking processes and offering valuable insights for industrial applications.

Keywords: hydrocracking; catalyst; neural networks; large language model

1. Introduction

Hydrocracking, a vital technology in petroleum refining, has the remarkable ability to convert heavy oils into lighter petroleum products, including light and heavy naphtha, as well as tailings [1]. This process is crucial for maximizing the value of heavy oils and meeting the increasing energy demand [2]. To obtain the desired hydrocarbons, one of the key factors is the use of high-performance hydrocracking catalysts [3–7]. To fully utilize the catalyst's performance and improve product properties, it is essential to match the catalyst with the feedstock's properties and the operating conditions. Currently, this process relies on experiment methods. Researchers have optimized key parameters of hydrogenation reactions over different catalysts using response surface methodology and central composite design, significantly enhancing the yield and selectivity of multiple products [8,9]. Unfortunately, this trial-and-error process is inefficient and costly.

Machine learning technology has shown great potential in optimizing catalytic processes. To optimize the operating conditions of hydrocracking units and improve the target product yield, Peng et al. established an

interpretable machine learning framework for analyzing their association, which can serve as a valuable reference for regulation in actual production [10]. Similarly, other researchers have established the relationship between catalyst structure and hydrocracking reaction performance using machine learning modeling, achieving rapid prediction of molecular sieve performance and identifying key determinants [11,12]. However, catalyst performance is mostly measured in terms of conversion or yield [13–15], which lacks the description of product properties, and there may be interactions between variables such as catalyst properties and operating conditions, so it is necessary to consider these variables comprehensively and systematically, so that researchers can be aware of the relationship between feedstocks, catalysts, operating conditions, and product distribution. Additionally, building data-driven models requires a certain level of specialized domain knowledge and coding skills, which may pose a significant challenge to researchers and limit broader adoption in industrial settings.

In recent years, artificial intelligence (AI) has been widely applied in the field of chemistry, changing the traditional mode of scientific research. AI accelerates the process of chemical research by searching chemical space, improving computational models, and providing support for the automation of experimental methods [16–18]. As a transformative tool in various fields, large language models (LLMs) have been quite successful in improving research efficiency. This is evident in applications such as literature data mining [19–21], assisting in robot design [22,23], and guiding experimental design [24–26]. Among them, prompt engineering explicitly communicates human ideas to LLMs through well-designed hints, guides LLMs to generate specific output content, and turns ideas into reality with significant efficiency [27,28], effectively overcomes knowledge gaps from different domains, and allows researchers to quickly get started without sufficient knowledge of related skills to speed up their research. Despite these advancements, current research lacks a comprehensive analysis that integrates catalyst properties, operating conditions, and product characteristics, hindering the further improvement of hydrocracking catalytic processes.

In this study, GPT-4 was used in combination with prompt engineering to assist in the development of neural network models for hydrocracking catalysts. The models utilized catalyst properties, operating conditions, and feedstock properties to predict the properties of hydrocracking tail oil. Relevant experimental data were collected from the laboratory of a Chinese refinery. After data cleaning and augmentation, the neural networks were constructed with the help of GPT-4 to capture the complex relationships among these factors accurately. The correlation was then interpreted by Grad-CAM visualization technology to reveal the key influencing factors. Finally, based on the excellent CNN model, the catalysts and operating conditions were optimized under the existing combination of conditions. This AI-assisted research on hydrocracking catalysts is expected to provide an efficient strategy for catalyst design and optimization of operating conditions, thereby improving overall process efficiency. The related workflow is illustrated in Figure 1.

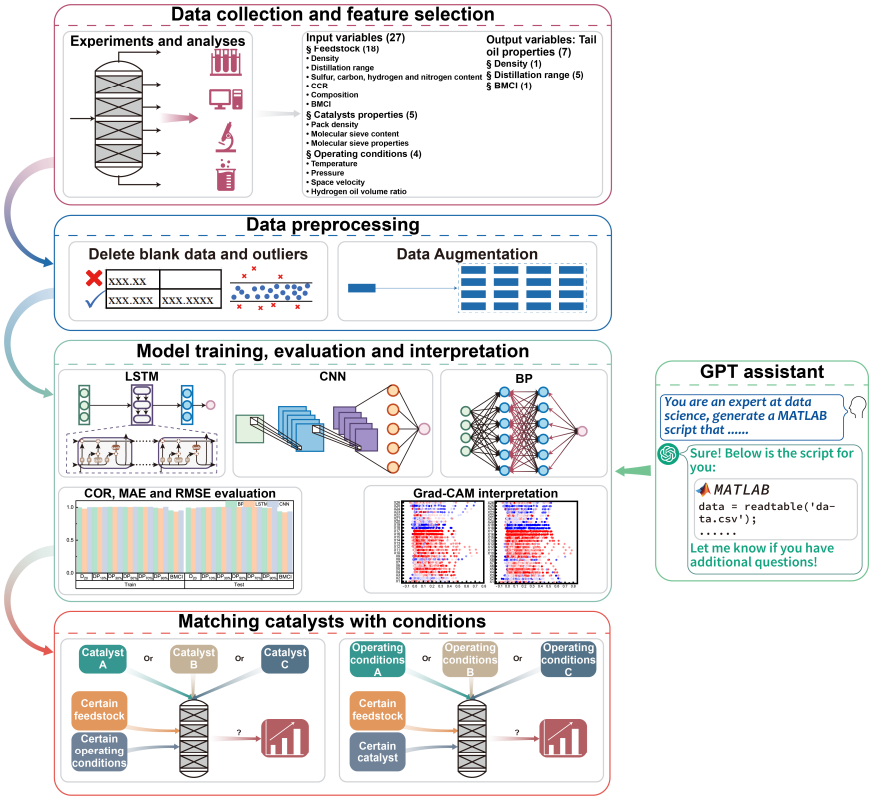

Figure 1. Flowchart of the content of this paper.

2. Methods

2.1. Data Preprocessing

A total of 43 sets of experimental data were collected and organized from the laboratory of a Chinese refinery, covering four brands of hydrocracking catalysts. To optimize the matching between catalysts, feedstocks, and operating conditions, a total of 27 features in three categories were selected as input variables based on the engineering experience, as detailed in Table 1. The properties of tail oil were chosen as the output variables, as listed in Table 2.

Table 1. Input features of ANNs.

Category	Variable	Abbreviation	Unit
Properties of catalyst	Packing density	ρ_{PD}	g/cm^3
	Si/Al ratio of molecular sieve	*SAR*	mol%
	Unit cell constant of molecular sieve	*UCC*	nm
	Relative crystallinity of molecular sieve	*RC*	%
	Molecular sieve content	C_{MS}	m%
Operating conditions	Space velocity	*SV*	h^{-1}
	Hydrogen-to-oil volume ratio	*HOVR*	-
	Reaction temperature	T_R	℃
	Reaction pressure	P_R	MPa
Properties of feedstock	Density, 20 °C	ρ_{20}^{F}	g/cm^3
	Distillation range, initial boiling point	DP_{IBP}^{F}	℃
	Distillation range, 10% distillation point	$DP_{10\%}^{F}$	℃
	Distillation range, 30% distillation point	$DP_{30\%}^{F}$	℃
	Distillation range, 50% distillation point	$DP_{50\%}^{F}$	℃
	Distillation range, 70% distillation point	$DP_{70\%}^{F}$	℃
	Distillation range, 90% distillation point	$DP_{90\%}^{F}$	℃
	Distillation range, 95% distillation point	$DP_{95\%}^{F}$	℃
	Carbon residue value	*CCR*	m%
	Bureau of mines correlation index	$BMCI^{F}$	-
	Sulfur content	M_S	m%
	Nitrogen content	M_N	m%
	Carbon content	M_C	m%
	Hydrogen content	M_H	m%
	Paraffin content	M_{CH}	m%
	Naphthene content	M_{CA}	m%
	Aromatic content	M_{AR}	m%
	Gum content	M_{GU}	m%

Table 2. Hydrocracking tail oil properties as output features of ANNs.

Variable	Abbreviation	Unit
Density, 20 °C	ρ_{20}^{TO}	g/cm^3
Distillation range, 10% distillation point	$DP_{10\%}^{TO}$	℃
Distillation range, 30% distillation point	$DP_{30\%}^{TO}$	℃
Distillation range, 50% distillation point	$DP_{50\%}^{TO}$	℃
Distillation range, 70% distillation point	$DP_{70\%}^{TO}$	℃
Distillation range, 90% distillation point	$DP_{90\%}^{TO}$	℃
Bureau of mines correlation index	$BMCI^{TO}$	-

After deleting the blank values, 27 groups of available data remained, which was a relatively small sample size. Considering that catalyst parameters such as ρ_{PD}, *SAR*, *UCC*, and *RC* fluctuate within specific ranges in industrial production, data augmentation was performed accordingly. Specifically, the ranges of these variables were determined based on long-term industrial production experience. For each group of experimental data, 20 random points within the determined ranges were selected to represent the catalyst properties. This approach expanded the original 27 groups of data to 540 data points, which was consistent with the non-uniform nature of catalyst properties in industrial reactors. The properties of each catalyst brand are detailed in Table 3. The dataset was then normalized and randomly divided into test and training sets in a 2:8 ratio.

Table 3. Range of catalyst features.

Brand	ρ_{PD} (g/cm^3)	*SAR* (mol%)	*UCC* (nm)	*RC* (%)	C_{MS} (m%)
A	0.75–0.85	9.0–12.0	2.436–2.440	≥85	50
B	>0.9	9.0–12.0	2.448–2.453	≥95	31
C	0.76–0.86	30–45	2.426–2.431	≥92	28.6
D	0.9–1.00	9–14	2.435–2.438	≥95	50

2.2. Prompt Engineering

In this study, prompt engineering was designed around five key aspects: role definition, task instruction, output specification, information supplementation, and interaction requirements. The role definition clarified the knowledge domains for GPT-4. Detailed and structured task instruction ensured a precise understanding of the researcher's intentions. Structured output specification enhanced the robustness and consistency of the results. Additional information provided necessary details and behavioral constraints. Furthermore, interaction requirements enabled it to make dynamic adjustments during tasks, preventing erroneous outputs caused by missing information.

Three types of prompt engineering were designed to address specific tasks in the construction and interpretation of neural network models in this paper. The first type of prompt engineering facilitated the generation of MATLAB scripts for training and evaluating neural network models, streamlining the process and ensuring reproducibility. The second type aided in creating MATLAB scripts for interpreting models using the Grad-CAM method, facilitating the understanding of the decision-making process of these models. The third type was utilized for interacting with GPT-4 to debug and refine the generated scripts. The full text of the prompt engineering is provided in the Supplementary Materials.

2.3. Neural Networks

Three neural networks were employed in this paper.

Backpropagation (BP), proposed by Rumelhart [29], is widely used for regression tasks, specifically numerical prediction. It consists of two main processes: the forward propagation generates the network output, while the backward propagation computes the error gradients and updates the weights using gradient descent to optimize the network's performance.

The long short-term memory (LSTM) network, introduced by Hochreiter [30], represents an improvement over traditional recurrent neural network (RNN). It modifies the structure of the hidden layer neurons and incorporates cell to store long-term information. This architecture enables LSTM to effectively manage long-term dependencies in sequential data, addressing the vanishing gradient problem that plagues traditional RNN.

LeNet, one of the pioneering convolutional neural networks (CNNs), was introduced by Yann LeCun for handwritten digit recognition [31]. The convolutional operation in CNN extracts local features to capture the spatial structure of the data effectively. Subsequently, the pooling operation reduces feature dimensions. This addresses the issues of excessive parameters, computational complexity, and difficulty in capturing local features in traditional neural networks for image recognition.

The neural networks were programmed using MATLAB R2023a, and their architectures are illustrated in Figure 2. All corresponding parameters are listed in Supplementary Materials Table S1. Model performance was evaluated using three metrics: the coefficient of determination (R^2), mean absolute error (*MAE*), and root mean square error (*RMSE*), with corresponding formulas provided in the supporting information.

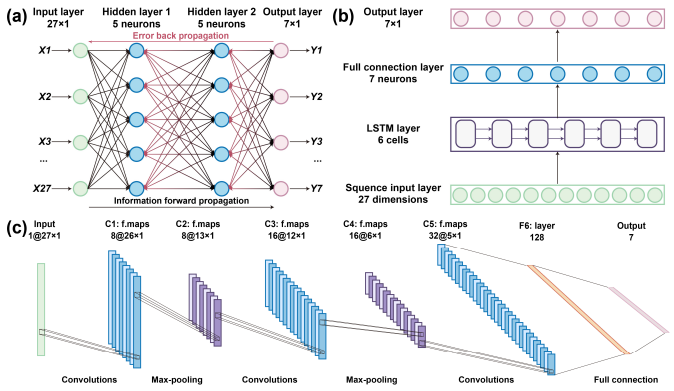

Figure 2. Architectures of neural networks used in this paper: (**a**) BP; (**b**) LSTM; (**c**) CNN.

2.4. Grad-CAM

Gradient-weighted class activation mapping (Grad-CAM), proposed by Ramprasaath R. Selvaraju [32], serves as a visualization technique to interpret the decision-making process of CNN. It assigns weights to each channel by calculating the global average pooling of gradients from the last convolutional layer, which subsequently generates heatmaps that highlight the most influential regions of the input image for the model's predictions. Grad-CAM can be directly applied to any trained CNN model without modifying the network structure or retraining the model, making it broadly applicable across various CNN architectures.

3. Results

3.1. Response from GPT-4

The prompt engineering was sent to GPT-4, and the corresponding responses are shown in Figure 3 and the supporting information. When interacting with GPT-4 for debugging, either branch 1 or 2 could be chosen to provide additional requirements to enhance the program or share runtime error messages for debugging purposes. Taking the initial script for interpreting CNN model using the Grad-CAM method as an example, as shown in Figure 3, GPT-4 assumed that the input features were images, whereas the actual CNN model used numerical matrices. Therefore, branch 1 was selected to provide GPT-4 with specific information about the input variables and the resulting response is shown in Supplementary Materials Figure S2. By generating the initial script using the first and second types of prompt engineering and then iteratively debugging with the third type, usable data-driven modeling scripts were readily available, lowering the technical barrier.

To generate a MATLAB script that accomplishes the tasks you described for using the Grad-CAM method on a CNN model, we will follow these steps:
1. Load the training set and CNN model from a MATLAB data file. You'll need to ensure the file path and names of the variables in the file are correctly specified in the script.
2. Execute Grad-CAM for each entry in the training set and for all output features (7 output features as mentioned).
3. Output the Grad-CAM results to a CSV file for each query point and each output feature.
Here's a MATLAB script template that accomplishes these tasks:

MATLAB

```
function interpret_cnn_with_gradcam()
    % Load data and model
    load('model_data.mat'); % Ensure 'model_data.mat'
contains 'trainedModel' and 'XTrain'

    % Assuming 'XTrain' is the training set and 'trained-
Model' is the CNN model
    % Verify that the model and data are loaded correctly
    disp("Model and training set loaded successfully.");

    % Prepare the CSV file to store Grad-CAM results
    resultsFileName = 'gradcam_results.csv';
    writematrix(["Index", "Feature", "Grad-CAM Heatmap
Data"], resultsFileName);

    % Execute Grad-CAM for each image and each output
feature
    numImages = size(XTrain, 4);
    numFeatures = 7; % Update if the number of features
changes
    for i = 1:numImages
        for j = 1:numFeatures
            heatmap = gradcam(trainedModel,
XTrain(:,:,:,i), "OutputFeature", j);

            % Process heatmap for output
            % This example assumes you convert the
heatmap to a string or numerical format that can be saved
to CSV
            heatmapData = extractHeatmapData(heatmap);

            % Append results to CSV
            dataToWrite = [i, j, heatmapData];
            writematrix(dataToWrite, resultsFileName,
'WriteMode', 'append');
        end
    end

    disp("Grad-CAM analysis completed and results
saved.");
end

function heatmapData = extractHeatmapData(heatmap)
    % This function should convert heatmap to a suitable
format for CSV
    % Example: Flatten and convert to string
    heatmapData = num2str(heatmap(:)');
end
```

Notes:
• Data and Model Loading: The script assumes the training data (XTrain) and model (trainedModel) are loaded from a file named model_data.mat. You need to make sure the actual variable names in the .mat file match those used in the script.
• Output Feature Specification: Grad-CAM is applied to each output feature individually. You should check how your model's output layers are structured to correctly specify the output features in the gradcam function call.
• Heatmap Processing: The function extractHeatmapData should be customized based on how you want to store the heatmap data. The example provided flattens the heatmap and converts it to a string for CSV storage, but you might need a different approach depending on your analysis needs.
This script provides a starting point. You may need to adapt the heatmap extraction and data processing parts based on your specific model structure and output requirements.

Figure 3. GPT-4 response for prompt engineering to assist in generating scripts for interpreting CNN using Grad-CAM.

3.2. Evaluation of Neural Networks

Figure 4 and Table 4 compare the evaluation metrics, including R^2, *MAE*, and *RMSE*, of three neural network models. As illustrated in Figure 4a, all three neural network models exhibit remarkable performance on both the training and testing datasets in terms of the R^2 metric. The minimum R^2 values for these networks on both datasets are observed during the prediction of $BMCI^{TO}$. Specifically, the R^2 values for BP, LSTM, and CNN on the training set are 0.9428, 0.9235, and 0.9418, respectively, while on the testing set, they are 0.9276, 0.9125, and 0.9263, respectively. Notably, all R^2 values exceed 0.9, indicating high accuracy across datasets.

Figure 4b,c further demonstrate the performance of the models in terms of the *MAE* and *RMSE*. Contrary to expectations, the prediction performance for $BMCI^{TO}$ is not the worst among the three models. Instead, the highest prediction accuracy is achieved when predicting ρ_{20}^{TO}. For BP, LSTM, and CNN, the *MAE* values on the training set are 0.0015, 0.0021, and 0.0010, respectively. On the testing set, they are 0.0016, 0.0018, and 0.0012, respectively. The *RMSE* values on the training set are 0.0019, 0.0031, and 0.0015, while on the testing set, they are 0.0020, 0.0026, and 0.0018, respectively. When predicting $BMCI^{TO}$, the three models exhibit relatively low error values. Specifically, the *MAE* values for BP, LSTM, and CNN are 0.2361, 0.2822, and 0.2348 on the training set, and 0.2695, 0.3185, and 0.2747 on the testing set. The *RMSE* values on the training set are 0.4185, 0.4839, and 0.4223, respectively. On the testing set, they are 0.4668, 0.5132, and 0.4710, respectively.

Among the five distillation variables of hydrocracking tail oil, the prediction accuracy for $DP_{90\%}^{TO}$ is the poorest. For BP, LSTM, and CNN, the *MAE* values on the training set are 6.3665, 8.2244, and 2.9499, respectively. On the testing set, they are 6.0013, 7.0472, and 3.4832, respectively. The *RMSE* values on the training set are 8.6014, 10.6196, and 4.1233, respectively. On the testing set, they are 8.2653, 9.3681, and 4.6981, respectively. In contrast, the prediction accuracy for $DP_{70\%}^{TO}$ is relatively higher. The *MAE* values on the training set are 3.6763, 3.8202, and 2.6921, respectively. On the testing set, they are 3.9613, 3.4817, and 3.0375, respectively. The *RMSE* values on the training set are 4.8593, 4.7343, and 3.7900, respectively. On the testing set, they are 5.0587, 4.3904, and 4.4353, respectively. Overall, as shown in Figure 4 and S3, the CNN model demonstrates the best performance among the three models.

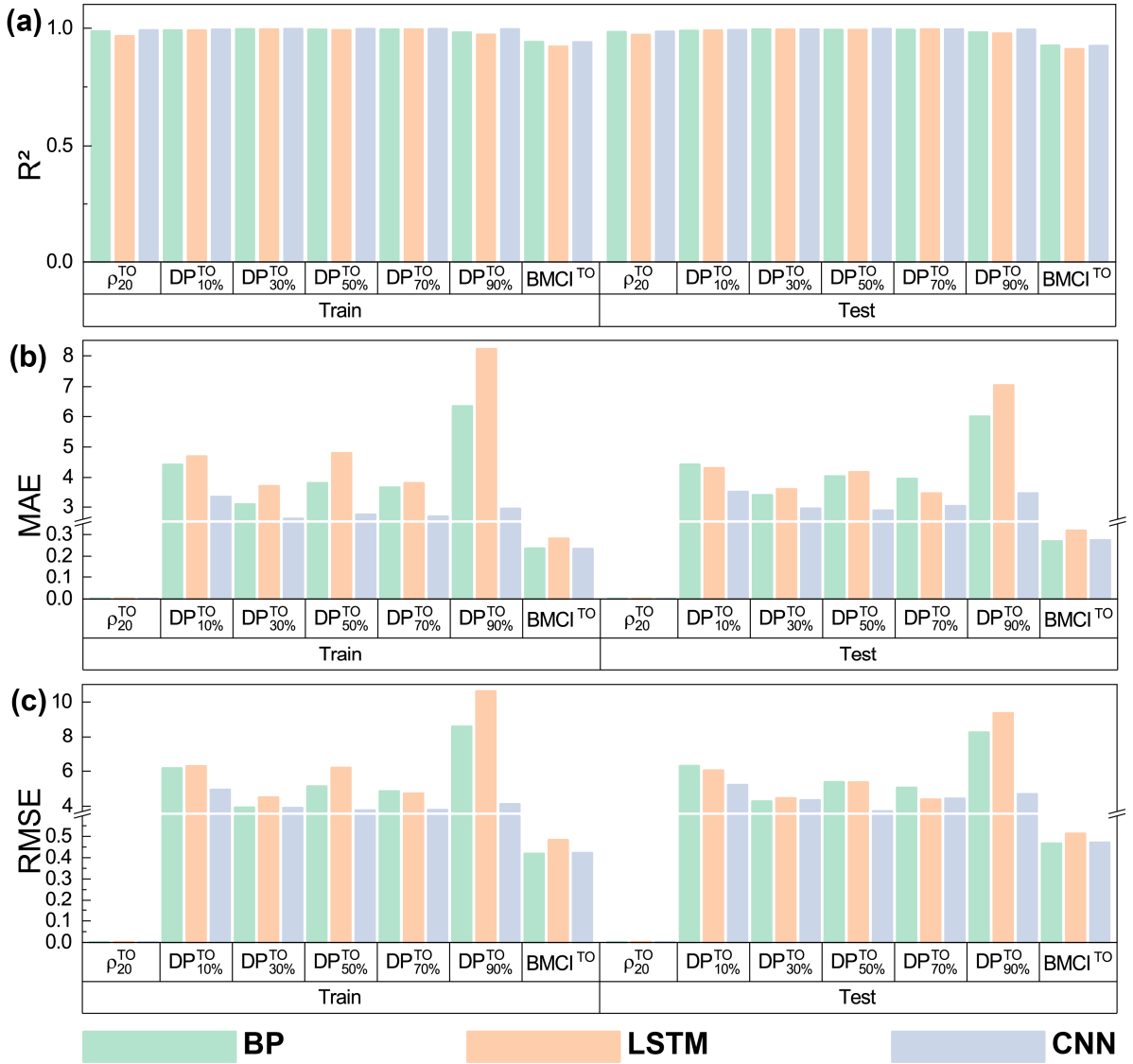

Figure 4. Comparison of neural network performance parameters: (**a**) R^2; (**b**) *MAE*; (**c**) *RMSE*.

Table 4. Comparison of R^2, *MAE*, and *RMSE* of different models on training and test sets.

Variable	Model	R^2(Train/Test)	*MAE*(Train/Test)	*RMSE*(Train/Test)
ρ_{20}^{TO}	BP	0.9874/0.9845	0.0015/0.0016	0.0019/0.0020
	LSTM	0.9674/0.9728	0.0021/0.0018	0.0031/0.0026
	CNN	0.9917/0.9863	0.0010/0.0012	0.0015/0.0018
$DP_{10\%}^{TO}$	BP	0.9916/0.9900	4.4204/4.4259	6.1683/6.3035
	LSTM	0.9913/0.9908	4.6942/4.3110	6.2905/6.0458
	CNN	0.9946/0.9931	3.3361/3.5338	4.9464/5.2209
$DP_{30\%}^{TO}$	BP	0.9968/0.9956	3.0931/3.3921	3.9279/4.2789
	LSTM	0.9957/0.9952	3.7213/3.6201	4.5102/4.4614
	CNN	0.9968/0.9955	2.6194/2.9519	3.9050/4.3465
$DP_{50\%}^{TO}$	BP	0.9947/0.9933	3.8189/4.0403	5.1396/5.3819
	LSTM	0.9923/0.9934	4.7997/4.1776	6.1959/5.3688
	CNN	0.9972/0.9968	2.7542/2.8873	3.7633/3.7165
$DP_{70\%}^{TO}$	BP	0.9951/0.9940	3.6763/3.9613	4.8593/5.0587
	LSTM	0.9953/0.9955	3.8202/3.4817	4.7343/4.3904
	CNN	0.9970/0.9954	2.6921/3.0375	3.7900/4.4353
$DP_{90\%}^{TO}$	BP	0.9824/0.9830	6.3665/6.0013	8.6014/8.2653
	LSTM	0.9732/0.9782	8.2244/7.0472	10.6196/9.3681
	CNN	0.9960/0.9945	2.9499/3.4832	4.1233/4.6981
$BMCI^{TO}$	BP	0.9428/0.9276	0.2361/0.2695	0.4185/0.4668
	LSTM	0.9235/0.9125	0.2822/0.3185	0.4839/0.5132
	CNN	0.9418/0.9263	0.2348/0.2747	0.4223/0.4710

3.3. Interpreting CNN with Grad-CAM

Figure 5 presents the Grad-CAM interpretation of the CNN.

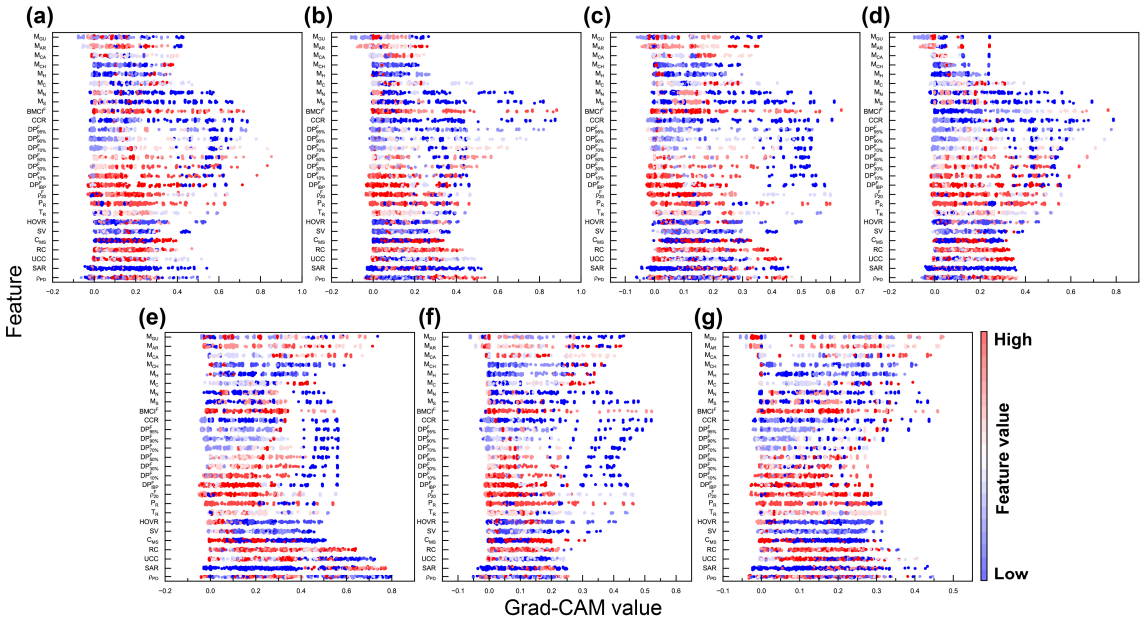

Figure 5. Grad-CAM interpretation of CNN for: (**a**) ρ_{20}^{TO}; (**b**) $DP_{10\%}^{TO}$; (**c**) $DP_{30\%}^{TO}$; (**d**) $DP_{50\%}^{TO}$; (**e**) $DP_{70\%}^{TO}$; (**f**) $DP_{90\%}^{TO}$; (**g**) $BMCI^{TO}$. A dot is made for the Grad-CAM value for each sample. The colors of the dots indicate the actual feature values, with red for higher values and blue for lower values.

The results indicate that the overall impact of the 27 input features on the seven properties of hydrocracking tail oil is predominantly positive, with relatively minor negative influences. Notably, the most significant negative impact is observed for the M_{GU} on the $DP_{10\%}^{TO}$ and $DP_{50\%}^{TO}$, reaching approximately −0.1. However, this negative influence remains small compared to the positive impact of M_{GU} on these properties. Moreover, different properties of hydrocracking tail oil are influenced by distinct primary factors. The ρ_{20}^{TO}, $DP_{10\%}^{TO}$, $DP_{30\%}^{TO}$, $DP_{50\%}^{TO}$, and $DP_{90\%}^{TO}$ are primarily affected by the properties of the feedstock, such as the distillation range, sulfur content, and nitrogen content. Both $\rho_{70\%}^{TO}$ and $BMCI^{TO}$ are mainly influenced by the composition of the feedstock and the properties of

the catalyst. The detailed primary influencing factors for each output feature are listed in Table 5, which ranks the input features based on their overall impact on the output features.

Table 5. The main influencing factors of output features.

Output Features	Main Influencing Factors
ρ_{20}^{TO}	$DP_{50\%}^{F}, DP_{30\%}^{F}, DP_{70\%}^{F}, DP_{10\%}^{F}, DP_{IBP}^{F}, DP_{90\%}^{F}, DP_{95\%}^{F}$, CCR, ρ_{20}^{F}, $BMCI^{F}$, P_R, T_R, M_N, M_S
$DP_{10\%}^{TO}$	$BMCI^{F}$, CCR, M_S, $DP_{95\%}^{F}, DP_{90\%}^{F}, DP_{70\%}^{F}$, M_N, ρ_{PD}
$DP_{30\%}^{TO}$	$BMCI^{F}$, CCR, M_S, M_N, ρ_{20}^{F}, P_R, T_R, $DP_{IBP}^{F}, DP_{10\%}^{F}, DP_{30\%}^{F}, DP_{50\%}^{F}, DP_{90\%}^{F}, DP_{95\%}^{F}$, $HOVR$
$DP_{50\%}^{TO}$	CCR, $DP_{95\%}^{F}$, $BMCI^{F}$, $DP_{90\%}^{F}, DP_{70\%}^{F}, DP_{50\%}^{F}$, M_S, $DP_{30\%}^{F}, DP_{10\%}^{F}, DP_{IBP}^{F}$, ρ_{20}^{F}, M_N, P_R, T_R
$DP_{70\%}^{TO}$	ρ_{PD}, SAR, UCC, M_{GU}, M_{AR}, M_{CA}, RC
$DP_{90\%}^{TO}$	$BMCI^{F}$, CCR, M_S, $DP_{95\%}^{F}, DP_{90\%}^{F}, \rho_{20}^{F}$, P_R, DP_{IBP}^{F}, T_R, $DP_{70\%}^{F}$, M_{GU}, M_{AR}, $DP_{10\%}^{F}, DP_{50\%}^{F}$, M_{CA}
$BMCI^{TO}$	M_{GU}, M_{AR}, $BMCI^{F}$, ρ_{PD}, M_{CA}, CCR, SAR, M_S, M_N, UCC, M_{CH}, M_C, M_H, $DP_{95\%}^{F}$

3.4. Optimization of Catalysts and Operating Conditions

The optimized CNN model can effectively and accurately predict product properties. Therefore, the preferred CNN model can be applied to optimize, screen and adjust the catalysts and operating conditions in the hydrocracking reaction. Specifically, we evaluated whether better hydrocracking tail oil properties could be achieved by either changing the catalyst brand or adjusting the operating conditions in two industrial settings. The first scene involved selecting an appropriate catalyst before commencing production. As shown in Figure 6a, using the feedstock and operating conditions associated with catalyst A, we randomly selected five combinations of catalyst features for catalysts B, C, and D, and the average of the CNN prediction results was taken to represent the potential tail oil properties for each catalyst grade. This step aimed to determine if replacing catalyst A with catalysts B, C, or D could yield improved outcomes. The second scene was optimizing operating parameters in daily production as the catalyst was secured. As shown in Figure 6b, using the feedstock and the same five combinations of catalyst features associated with catalyst D, we replaced the operating conditions with those from catalysts A, B, and C, respectively, and again averaged the predicted results from the CNN model to represent the tail oil properties under these adjusted operating conditions. This analysis assessed whether modifying the operating conditions could enhance the tail oil properties when using catalyst D.

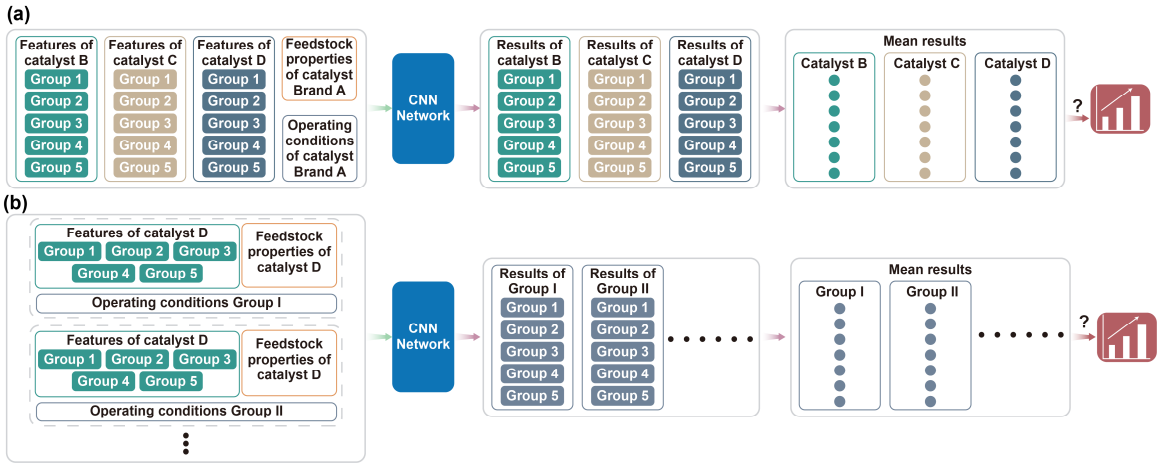

Figure 6. Flowchart of this section for: (**a**) the question of changing the brand of catalyst; (**b**) the question of changing operating conditions.

Figure 7 presents the predicted values for hydrocracking tail oil properties when replacing catalyst A with catalysts B, C, and D under constant operating conditions and feedstock properties. After changing the catalyst, the predicted ρ_{20}^{TO} values for condition groups 1, 2, 6, and 7 decrease compared to the experimental values, while those for condition groups 3, 4, and 5 increase. Lower ρ_{20}^{TO} indicates a higher content of light components, making it more suitable as a feedstock for cracking to ethylene. Overall, catalyst B consistently yield the lowest ρ_{20}^{TO} among the three new catalysts in most condition groups. The changes in the 10% to 90% distillation points of

hydrocracking tail oil after changing the catalyst are shown in Figure 7b–f, while the changes in distillation range are shown in Figure 7h, calculated using Equation (1).

$$Distillation\ range = DP_{90\%}^{TO} - DP_{10\%}^{TO} \quad (1)$$

Overall, except for the results of condition group 4 with all catalysts and condition groups 6 and 7 with catalyst D, using catalysts B, C, and D generally increases $DP_{10\%}^{TO}$ and decreases $DP_{90\%}^{TO}$, narrowing the distillation range. The narrower distillation range indicates a higher proportion of light components in the hydrocracking tail oil, which is more suitable for cracking to ethylene. The changes in the $BMCI^{TO}$ after changing the catalyst are shown in Figure 7g. Except for the results of condition group 3 with catalysts B and C and condition groups 6 and 7 with catalyst D, using catalysts B, C, and D generally reduces the $BMCI^{TO}$. Considering the density, distillation range, and *BMCI* of hydrocracking tail oil, changing to catalyst B can produce hydrocracking tail oil that is better suited for cracking to ethylene under the operating conditions and feedstock properties associated with catalyst A.

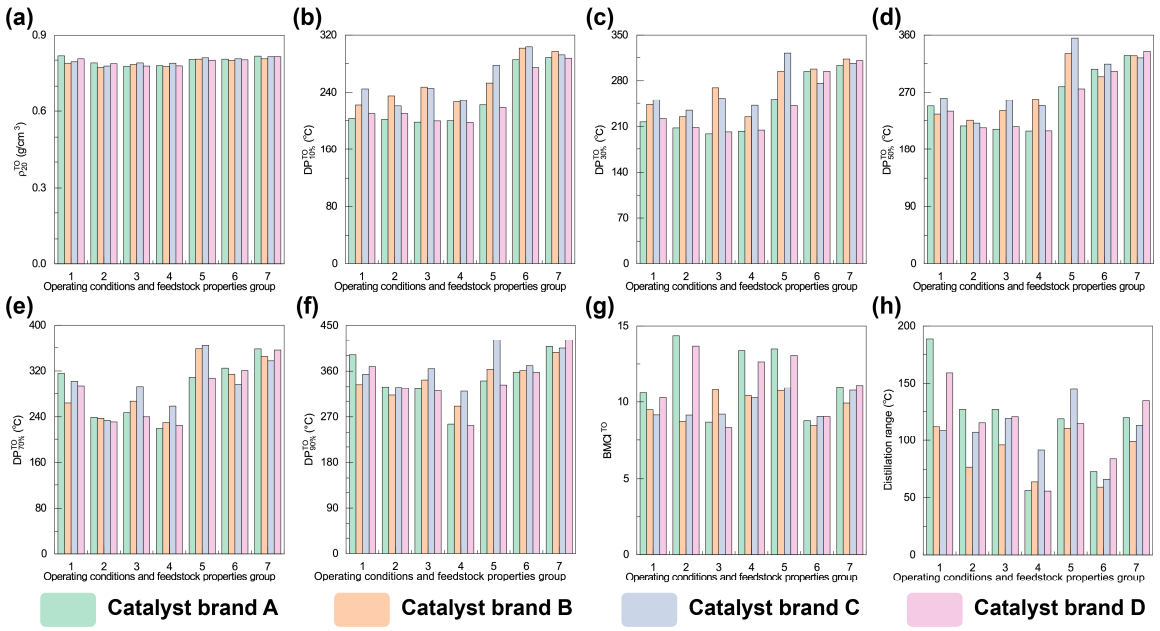

Figure 7. The predicted values of CNN for each output feature when changing catalyst brands: (**a**) ρ_{20}^{TO}; (**b**) $DP_{10\%}^{TO}$; (**c**) $DP_{30\%}^{TO}$; (**d**) $DP_{50\%}^{TO}$; (**e**) $DP_{70\%}^{TO}$; (**f**) $DP_{90\%}^{TO}$; (**g**) $BMCI^{TO}$; (**h**) Distillation range.

Figure 8 illustrates the predicted values for hydrocracking tail oil properties as operating conditions are varied. The results indicate that the changes in the properties of hydrocracking tail oil resulting from altered operating conditions are more complex and less consistent. As shown in Figure 8h, only condition groups 2 and 5 result in a narrower distillation range. However, as indicated in Figure 8g, only condition group 5 among these two can reduce the $BMCI^{TO}$. Figure 8a shows that eight condition groups can reduce the density ρ_{20}^{TO}, while Figure 8g indicates that twelve condition groups can lessen the $BMCI^{TO}$. These findings suggest that optimizing operating conditions is more challenging than changing the catalyst brand, and that simply searching for the best conditions among existing combinations is insufficient to find the optimal operating conditions.

In summary, these analyses not only provide valuable insights into the complex relationship between catalyst properties, operating conditions, and hydrocracking tail oil properties but also establish a critical foundation for future quantitative research.

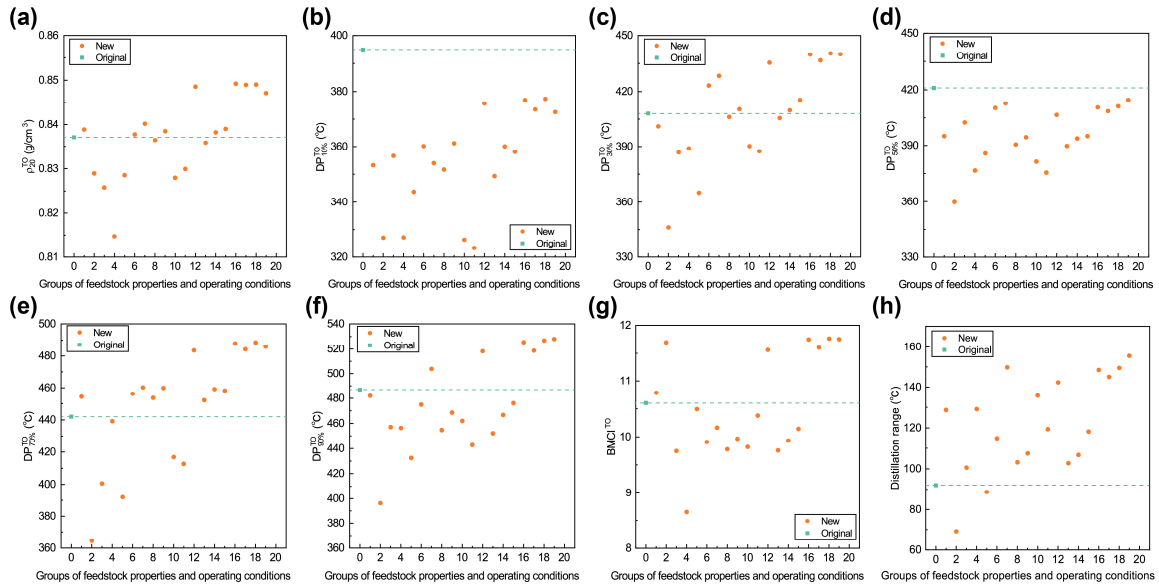

Figure 8. The predicted values of CNN for each output feature when changing operating conditions: (**a**) ρ_{20}^{TO}; (**b**) $DP_{10\%}^{TO}$; (**c**) $DP_{30\%}^{TO}$; (**d**) $DP_{50\%}^{TO}$; (**e**) $DP_{70\%}^{TO}$; (**f**) $DP_{90\%}^{TO}$; (**g**) $BMCI^{TO}$; (**h**) Distillation range.

4. Discussion

To conveniently build data-driven models for hydrocracking, AI-assisted training and evaluation of neural networks were performed through well-designed prompt engineering. Additionally, the Grad-CAM method was utilized to interpret the CNN model, revealing the specific impacts of different input features on hydrocracking tail oil.

The industrial experimental data were collected from a Chinese refinery, with catalyst properties, feedstock properties, and operating conditions selected as input features. The dataset was expanded through random sampling within the reasonable range of catalyst properties. The CNN model predicted hydrocracking tail oil properties with the highest accuracy. Based on the CNN model, the effects of changing catalysts and adjusting operating conditions on the tail oil properties were qualitatively discussed, providing theoretical support for industrial practice. Although the adjustment of operating conditions somewhat improved the properties of tail oil, the trends of tail oil properties are more complicated. This is expected to further combine more optimization algorithms and process knowledge for fine tuning.

Due to limited data, the current model faces a potential risk of overfitting, which may compromise its predictive accuracy. However, the model still demonstrates the significant potential of AI-assisted catalyst development and application. The expansion of high-quality data is crucial for model training. In the future, we will expand the dataset to enhance the accuracy and generalizability of the model, thereby promoting the optimization of the hydrocracking process.

Supplementary Materials

The additional data and information can be downloaded at: https://media.sciltp.com/articles/others/25102214233057 54/SCE-2508000049-Supplementary-Materials-FC-done.pdf.

Author Contributions

S.L.: Conceptualization (supporting), data curation (lead), investigation (equal), methodology (equal), software (lead), validation (lead), visualization (supporting), writing—original draft (lead), writing—review and editing (equal); Z.P.: Conceptualization (supporting), data curation (supporting), investigation (equal), methodology (equal), software (supporting), validation (supporting), visualization (lead), writing—original draft (supporting), writing—review and editing (supporting); C.L.: Conceptualization (equal), funding acquisition (equal), methodology (equal), project administration (equal), supervision (equal), writing—review and editing (supporting); C.P.: Data curation (supporting), conceptualization (equal), funding acquisition (equal), methodology (equal), project administration (equal), supervision (equal); X.F.: Data curation (supporting), conceptualization (equal); H.L.: funding acquisition (equal), methodology (equal). All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the National Key Research and Development Program of China (No. 2023YFA1507601), National Natural Science Foundation of China (No. 22278127 and No. 22378038), the Fundamental Research Funds for the Central Universities (No. 2022ZFJH004), the Shanghai Pilot Program for Basic Research (22T01400100-18), and the Natural Science Foundation of Liaoning Province, China (No. 2024-MSBA-15).

Data Availability Statement

The authors do not have permission to share the data that used for training and testing the neural networks. Other data are available on request.

Conflicts of Interest

The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies

During the preparation of this work, the authors utilized GPT-4 as an AI assistant to facilitate the development and interpretation of data-driven models for hydrocracking catalysis. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

1. Peng, C.; Fang, X.; Zeng, R. Research and Development of Hydrocracking Catalysts and Technology. In *Catalysis*, Spivey, J., Dooley, K.M., Han, Y.F., Eds.; The Royal Society of Chemistry: London, UK, 2016; Volume 28, pp. 86–118.
2. Fukuyama, H.; Terai, S.; Uchida, M.; et al. Active carbon catalyst for heavy oil upgrading. *Catal. Today* **2004**, *98*, 207–215.
3. Peng, C.; Zhou, Z.; Cheng, Z.; et al. Upgrading of light cycle oil to high-octane gasoline through selective hydrocracking over non-noble metal bifunctional catalysts. *Energy Fuels* **2019**, *33*, 1090–1097.
4. Sun, J.A.; Selvam, E.; Bregvadze, A.; et al. Hydrocracking of polyolefins over ceria-promoted Ni/BEA catalysts. *Green Chem.* **2025**, *27*, 3905–3915.
5. Vance, B.C.; Yuliu, Z.; Najmi, S.; et al. Unlocking naphtha from polyolefins using Ni-based hydrocracking catalysts. *Chem. Eng. J.* **2024**, *487*, 150468.
6. Yan, J.; Li, G.; Lei, Z.; et al. Upcycling polyolefins to methane-free liquid fuel by a Ru_1-ZrO_2 catalyst. *Nat. Commun.* **2025**, *16*, 2800.
7. Zhan, J.; Li, L.; Dai, R.; et al. Engineering porous beta zeolite-encapsulated nickel catalyst for waste polyolefins upcycling. *Appl. Catal. B Environ.* **2025**, *373*, 125359.
8. Kristensen, T.; Hulteberg, C.; Blomberg, S.; et al. Parametric analysis and optimization of vanillin hydrodeoxygenation over a sulfided Ni-Mo/δ-Al_2O_3 catalyst under continuous-flow conditions. *Top. Catal.* **2023**, *66*, 1341–1352.
9. Li, X.; Wang, Q.; Wu, Y.; et al. Optimization of key parameters using RSM for improving the production of the green biodiesel from FAME by hydrotreatment over Pt/SAPO-11. *Biomass Bioenergy* **2022**, *158*, 106379.
10. Pang, Z.; Huang, P.; Lian, C.; et al. Data-driven prediction of product yields and control framework of hydrocracking unit. *Chem. Eng. Sci.* **2024**, *283*, 119386.
11. Ma, Q.; Nie, H.; Yang, P.; et al. Insights into structure-activity relationships between Y zeolites and their n-C_{10} hydrocracking performances via machine learning approaches. *Chin. J. Catal.* **2025**, *71*, 187–196.
12. Wang, W.; Li, M.; Zhang, Y.; et al. Structure-performance relationship between zeolites properties and hydrocracking performance of tetralin over NiMo/Al_2O_3-Y catalysts: A machine-learning-assisted study. *Fuel* **2025**, *390*, 134652.
13. Oberhausen, C.M.; Auchenbach, K.E.; Vlachos, D.G. Investigating the role of acid sites in the hydrocracking of polyethylene-EVOH multilayer film waste over Pt/BEA Catalyst. *Chem. Eng. J.* **2025**, *508*, 160869.
14. Wang, J.; Yan, J.; Cui, Q.; et al. Effect of SiO_2 support particle sizes on the performance of FeZn catalysts in VR slurry-phase hydrocracking. *Catal. Today* **2025**, *449*, 115183.
15. Zhang, K.; Hu, Z.; Ren, L.; et al. Research on the process of naphtha hydrocracking to chemical materials. *Carbon Resour. Convers.* **2025**, *8*, 100315.
16. Mroz, A.M.; Basford, A.R.; Hastedt, F.; et al. Cross-disciplinary perspectives on the potential for artificial intelligence across chemistry. Chem. Soc. Rev. **2025**, *54*, 5433–5469.
17. Tkatchenko, A. Machine learning for chemical discovery. *Nat. Commun.* **2020**, *11*, 4125.
18. Yang, L.; Guo, Q.; Zhang, L. AI-assisted chemistry research: A comprehensive analysis of evolutionary paths and hotspots through knowledge graphs. *Chem. Commun.* **2024**, *60*, 6977–6987.

19. Fu, Z.; Huang, P.; Wang, X.; et al. Artificial intelligence-assisted ultrafast high-throughput screening of high-entropy hydrogen evolution reaction catalysts. *Adv. Energy Mater.* **2025**, 2500744.
20. Wei, C.; Shi, Y.; Mu, W.; et al. Large language models assisted materials development: Case of predictive analytics for oxygen evolution reaction catalysts of (Oxy)hydroxides. *ACS Sustain. Chem. Eng.* **2025**, *13*, 5368–5380.
21. Zheng, Z.; Florit, F.; Jin, B.; et al. Integrating machine learning and large language models to advance exploration of electrochemical reactions. *Angew. Chem. Int. Ed.* **2025**, *64*, e202418074.
22. Stella, F.; Della Santina, C.; Hughes, J. How can LLMs transform the robotic design process? *Nat. Mach. Intell.* **2023**, *5*, 561–564.
23. Vemprala, S.H.; Bonatti, R.; Bucker, A.; et al. ChatGPT for robotics: Design principles and model abilities. *IEEE Access* **2024**, *12*, 55682–55696.
24. Jablonka, K.M.; Schwaller, P.; Ortega-Guerrero, A.; et al. Leveraging large language models for predictive chemistry. *Nat. Mach. Intell.* **2024**, *6*, 122–123.
25. Su, Y.; Wang, X.; Ye, Y.; et al. Automation and machine learning augmented by large language models in a catalysis study. *Chem. Sci.* **2024**, *15*, 12200–12233.
26. Zheng, Z.; Rong, Z.; Rampal, N.; et al. A GPT-4 reticular chemist for guiding MOF discovery. *Angew. Chem. Int. Ed.* **2023**, *62*, e202311983.
27. Chen, B.; Zhang, Z.; Langrené, N.; et al. Unleashing the potential of prompt engineering for large language models. *Patterns* **2025**, *6*, 101260.
28. Luo, F.; Zhang, J.; Wang, Q.; et al. Leveraging prompt engineering in large language models for accelerating chemical research. *ACS Cent. Sci.* **2025**, *11*, 511–519.
29. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536.
30. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
31. Lecun, Y.; Bottou, L.; Bengio, Y.; et al. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324.
32. Selvaraju, R.R.; Cogswell, M.; Das, A.; et al. Grad-CAM: Visual Explanations from Deep Networks Via Gradient-Based Localization. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.