THE JOURNAL OF PHYSICAL CHEMISTRY C
A JOURNAL OF THE AMERICAN CHEMICAL SOCIETY

Editors' Choice
Open Access

pubs.acs.org/JPCC

Article

A Machine Learning and Explainable AI Framework Tailored for Unbalanced Experimental Catalyst Discovery

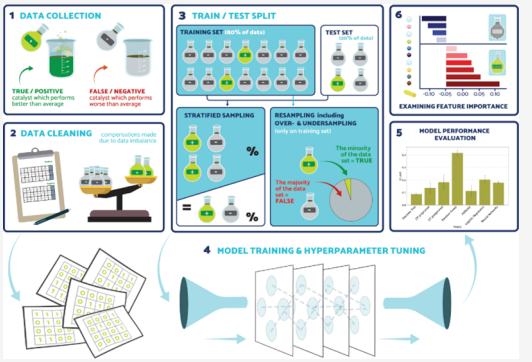
Parastoo Semnani,* Mihail Bogojeski, Florian Bley, Zizheng Zhang, Qiong Wu, Thomas Kneib, Jan Herrmann, Christoph Weisser, Florina Patcas, and Klaus-Robert Müller*

Cite This: J. Phys. Chem. C 2024, 128, 21349−21367

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: The successful application of machine learning (ML) in catalyst design has been made difficult by the challenges associated with collecting high-quality and diverse data. Due to the complex interactions between catalyst components, the design of novel catalysts has long relied on trial-and-error, a costly and labor-intensive process that results in scarce data that is heavily biased toward undesired, low-yield catalysts. Such data presents a challenge for training ML models that generalize well to novel compositions, which is necessary for the success of ML-guided catalyst discovery. Despite the growing popularity of ML applications in this field, most efforts so far have not focused on dealing with the challenges presented by such experimental data. In this work, we introduce a robust ML and explainable artificial intelligence (XAI) framework that incorporates a series of well-established ML methods designed to improve model performance and provide reliable evaluations for catalytic yield classification in the context of scarce and class-imbalanced data. We apply this framework to classify the yields of different catalyst combinations in the oxidative coupling of methane reaction and use it to evaluate the performance of a range of ML models: tree-based models (such as decision trees, random forest, and gradient boosted trees), logistic regression, support vector machines, and neural networks. Our experiments demonstrate that the methods used in our framework lead to more robust performance estimates and reduce the effect of class imbalance on model training, resulting in significant improvements in the predictive capability of all but one of the evaluated models. Additionally, the XAI component of the framework analyzes the decision-making process of each ML model by identifying the most important features for predicting catalyst performance. Our analysis found that XAI methods that provide class-aware explanations, such as Layer-wise Relevance Propagation, managed to identify key components that contribute specifically to high-yield catalysts. These findings align with chemical intuition and existing literature, reinforcing their validity. We believe this framework can serve as a blueprint and a set of best practices for ML applications in catalysis, driving future research while delivering robust models and actionable insights that can assist chemists in designing and discovering novel catalysts with superior performance.

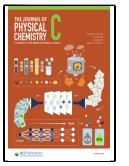


INTRODUCTION

Machine learning (ML) models have recently become popular in the field of heterogeneous catalyst design.[1−10] The inherent complexity of the interactions between catalyst components is very high, leading to both synergistic and antagonistic effects on catalyst yield that are difficult to disentangle. Therefore, the discovery of well-performing catalysts has long relied on serendipitous trial and error.[11,12]

Unlike traditional methods based on simplified models and heuristics, ML methods excel at identifying complex patterns and nonlinear relationships between various catalyst components. This capability is particularly advantageous in catalyst design, where ML can offer insights into nuanced component interactions, crucial for optimizing yield.[13−15] However, the application of ML methods in catalyst design faces several challenges, with the most prominent challenge being the scarcity of large and unbiased data sets. Despite significant efforts in data acquisition and curation,[16−20] data sets often remain small and limited due to the high costs in labor and time. Additionally, the choice of catalysts, elements and supports can cause certain types of biases to manifest in the data in two key ways. First, existing data often favors historically successful or easily testable catalysts.[21−23] This leads to a bias in the selection of elements and compounds and can lead to an over-representation of certain components, most commonly elements and supports that have historically been more accessible for testing and have been a part of successful

Received: August 7, 2024
Revised: November 18, 2024
Accepted: November 18, 2024
Published: December 6, 2024



catalysts. However, such elements and supports only occupy a very narrow spectrum of all possible compounds, leaving many other elements and supports underrepresented or not represented at all in the data.[24] This type of bias is difficult to address with machine learning, since a large part of the compound space is not present in the training data, making it impossible for machine learning models to extrapolate and learn the interactions between the elements and supports that are not represented, regardless of the size of the data set. However, this bias in the selection of elements and supports can be addressed when curating the data set itself, as shown by the effort in Nguyen et al.,[16] where the bias in the selection of catalyst components was addressed by explicitly performing a randomized and unbiased selection of elements and supports.

Nevertheless, data sets with a more unbiased selection of elements and supports can often lead to a second type of bias in the data, this time in terms of the representation of low- and high-yield catalysts. Due to the complexity of the interactions between catalyst components, a randomized unbiased selection of catalyst components is much more likely to result in a low-yield catalyst, making high-yield catalysts inherently much rarer and underrepresented. This highlights a fundamental challenge: while an unbiased data set might provide a broader exploration of the compound space, it often includes many suboptimal catalysts. In our scenario, this results in an imbalance in the two class labels, which can pose a challenge when training and evaluating machine learning models. Despite this, these problems can be mitigated by appropriate use of machine learning techniques, which is precisely one of the focal points of this work.

To tackle these challenges, we propose a robust ML and explainable AI (XAI) framework designed to handle the scarcity and imbalance of experimental catalyst data (see Figure 2). The primary goal of this framework is to offer a conceptual blueprint aimed at establishing best practices for robust evaluation and analysis of machine learning models, particularly when working with such challenging experimental data sets.

The framework is composed on a series of ML methods that are commonly used for dealing with scarce and imbalanced data, such as nested cross-validation, stratified sampling, resampling, and various XAI methods, all combined into a unified workflow. Nested-cross validation is used to obtain robust performance estimates despite the variability inherent when working with small data sets, while the sampling methods mitigate biases and the impacts of over-represented classes during training. Additionally, we utilize the F1-score, a performance measure that is uniquely well-suited to our problem setting, which is crucial for correctly evaluating the predictive capability of any model.

Building on this foundation, this study further contributes to ML-guided catalyst design by applying our framework on an unbiased data set for the oxidative methane coupling (OCM) reaction, which includes a diverse selection of elements and supports, introduced by Nguyen et al.[16] To systematically assess the effectiveness of our framework, we use it to train and evaluate a variety of ML models on the aforementioned OCM data set, and document the changes in performance resulting from the various framework components.

Recognizing the necessity for model interpretability in catalysis, we also apply XAI methods[25−30] to analyze strongly nonlinear models, such as neural networks and support vector machines (SVM), identifying key features that influence their decisions and providing insights into their decision-making processes. This enables us to determine which catalyst components have the strongest contribution toward the model's prediction. This information enables us to develop a generative model designed to predict potential high-yield catalyst candidates.

In summary, this work proposes more robust performance metrics and sampling strategies, explores a diverse set of ML models, and applies XAI methods to analyze their decisions, and disentangle contributions of each component to high-yield catalysts. We aim to pave the way for effective ML-guided catalyst design under data scarcity, providing a blueprint and best practices that can improve future ML efforts for more efficient experimental design and accelerate catalyst discovery.

MATERIALS AND METHODS

Data. Under the effects of certain catalysts, OCM converts methane to C_2 products, e.g., C_2H_4 and C_2H_6, which serve as the fundamental building blocks in the chemical industry. Thus, the effectiveness of a catalyst is often measured by the percentage of C_2 yield. Researchers have applied catalyst informatics to OCM, using data analysis and ML methods to identify synergistic combinations like Na−La, Na−Mn, and Ba−Sr.[18] Current challenges include inconsistent experimental methods and biases in component choices among different publications.[22,31]

To address these challenges, Nguyen et al.[16] have gathered unbiased and process-consistent OCM data via a high-throughput screening (HTS) instrument for 300 quaternary structured catalysts, with each component being randomly selected from a predefined range of candidates. The quaternary structure of the catalyst, M1-M2-M3/support, consists of three active elements (M1-M2-M3) randomly selected from 28 commonly used elements (including "none" as an option) with replacement, and one support randomly selected from 9 oxides. To ensure unbiased selection, 300 combinations are randomly chosen as candidate catalysts from all possible combinations. Evaluation experiments for each candidate catalyst under 135 different reaction conditions are then performed via HTS. Specifically, one combination of temperature, input ratios, total flow, and atmospheric pressure defines one reaction condition. Only the data of one reaction condition with the highest C_2 yield is recorded for each candidate catalyst. Apart from C_2 yield, another two quantities CH_4 conversion and C_2 selectivity are recorded. CH_4 conversion measures how much of the input methane is converted. C_2 selectivity measures how much of the output is the desired output, i.e., C_2 products. Thus, the product of CH_4 conversion and C_2 selectivity equals C_2 yield, which also indicates the conversion-selectivity trade-off. There are, in total, 291 records for individual catalysts in the data set, since the performance scores of 9 catalysts are missing. Nguyen et al.[16] provides informative interpretations from a chemical perspective based on the statistical analysis of the experimental data.

To facilitate the efficient discovery of combinatorial catalysts, Nguyen et al.[16] have prepared the data with an unbiased selection of elements, making it potentially beneficial for ML applications. The target variable is set as the best C_2 yield, which is a binary variable that is set as true if the yield is larger than 13% and false when the yield is lower. The data set consists of 51 high-yield catalysts and 240 low-yield catalysts in total. The data consists of 49 Boolean features denoting the

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

presence of elements (27), supports (9), and periodic table groups (13) in a given catalyst combination. Due to its diverse and bias-free construction, we chose to use this data set as an example for our proposed framework for training, evaluation, and explainable AI.

While we strongly appreciate the efforts of Nguyen et al.[16] in curating this unbiased data set and making it publicly available, it is essential to highlight certain characteristics and potential issues of this data to provide context for our analysis.

First, we have found that the features denoting whether an element belongs to a specific group in the periodic system are superfluous, as they do not seem to improve the overall performance of the models when accounting for class imbalance. Additionally, they make it more difficult to disentangle feature importance attribution from explainability methods since they correlate strongly with the elements belonging to the group. This is especially the case with groups 3, 5, 7, 8, 9, 11, and 12, which have only 1 element each, respectively Y, V, Mn, Fe, Co, Cu, and Zn, resulting in the corresponding features being fully correlated. We note that this effect is specific to the current data set due to the sparse nature of the feature representation and selection of elements. In the case of larger and more diverse data sets, where a broader range of elements are included, such main group information may prove to be a very useful feature. A more detailed discussion on this can be found in Supplementary Section Effect of periodic table group features.

Finally, it is important to note that the data set only includes the optimal operation conditions for the catalyst material. As a consequence, the test conditions, such as temperature and Gas Hourly Space Velocity, serve as identifiers for each data instance rather than features that can be used for training and analysis. Additionally, Nguyen et al.[16] highlight the process sensitivity of the OCM reaction, indicating that test conditions may have a more profound impact on catalyst performance than changes in material composition.

In supervised ML, the overarching goal is to develop models that generalize well to unseen data. However, the exclusive use of optimal test conditions within the data set poses a challenge to the model's generalization ability. By training solely on data characterized by optimal process conditions, the models may struggle to accurately predict the target variable for unseen cases subject to different sets of process conditions.

However, this does not invalidate the approach of focusing on the best yield of a catalyst composition given a set of processes condition. For instance, if the yield of a particular combination remains below the desired threshold and is classified as low, it discourages further testing, as the potential for success is limited. On the other hand, if the predicted yield falls within the high-yield category, conducting a series of tests under varying process conditions can help identify the optimal conditions for achieving the desired yield. This method still offers significant cost savings compared to trial-and-error-based high-throughput screening, since only a limited set of promising compositions need to be tested. The reduction in cost would be very similar to that of models trained on data with specific process conditions, since experimental validation remains necessary to confirm whether the predicted optimal conditions for each candidate composition hold true in practice.

Performance Evaluation Metrics. Among the various measures used to evaluate the performance of ML models, this study focuses on accuracy, precision, recall, and F1-score. These metrics measure different aspects of model performance, each suited to different objectives and contexts. The equations for all the measures we use in this work are shown in Table 1, and an illustration comparing the relationships between the performance measures can be found in Figure 1.

Table 1. Definition of Different Commonly Used Performance Evaluation Metrics for ML Models[a]

$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$	$\text{Precision} = \frac{TP}{TP + FP}$
$\text{Recall} = \frac{TP}{TP + FN}$	$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$

[a]TP denotes the number of true positives, TN that of true negatives, FP denotes the number of false positives, and FN the number of false negatives.

Accuracy is one of the most widely used evaluation metrics for ML models, which measures the proportion of correct predictions, encompassing both true positives and true negatives in a single metric. It is particularly suitable for balanced data sets, where the number of samples in each class is roughly equal. In the case of highly unbalanced class ratios, accuracy can be misleading since a classifier only predicting the majority class would still be able to achieve high accuracy.

To overcome this shortcoming, other evaluation measures have been introduced that better reflect the different aspects of the problem. In the case of catalyst design, we are more interested in one of the two classes, namely high-yield catalysts. This is why the measures precision, recall and F1-score are especially relevant here. Precision measures the proportion of true positive predictions among all positive predictions and is valuable when the cost of false positives is high. Recall, also known as sensitivity or true positive rate, gauges the ability of the model to capture all positive samples and is crucial when the cost of false negatives is a concern. It is defined as the ratio of true positives to the sum of true positives and false negatives. Finally, the F1-score is the harmonic mean of precision and recall and is particularly useful when the class ratios are imbalanced, and the positive class is especially important, which is the case in our catalyst yield classification task.

Resampling. ML models often struggle within scenarios with highly imbalanced class distributions.[32,33] Because most ML models are designed for data sets with an equal number of observations for each class, if the imbalance is not accounted for, the models may prioritize the majority class and overlook the minority class, negatively impacting overall performance.

One common approach to addressing this issue is by employing resampling techniques, which use various strategies to oversample the minority class or undersample the majority class in order to balance the data set.[34−41]

We choose to perform oversampling using Synthetic Minority Oversampling Technique (SMOTE),[42] and following the recommendation of Chawla et al.[42] we combine SMOTE with random undersampling of the majority class. We opted for SMOTE because it is a robust and well-established method that has been effectively used for many years, demonstrating consistent performance advantages over alternative resampling techniques. Its ability to generate synthetic samples helps improve model generalization without the risks associated with simple duplication or excessive reduction of the majority class.

Cross-Validation. When dealing with small data sets, the performance of the model can depend quite strongly on the

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

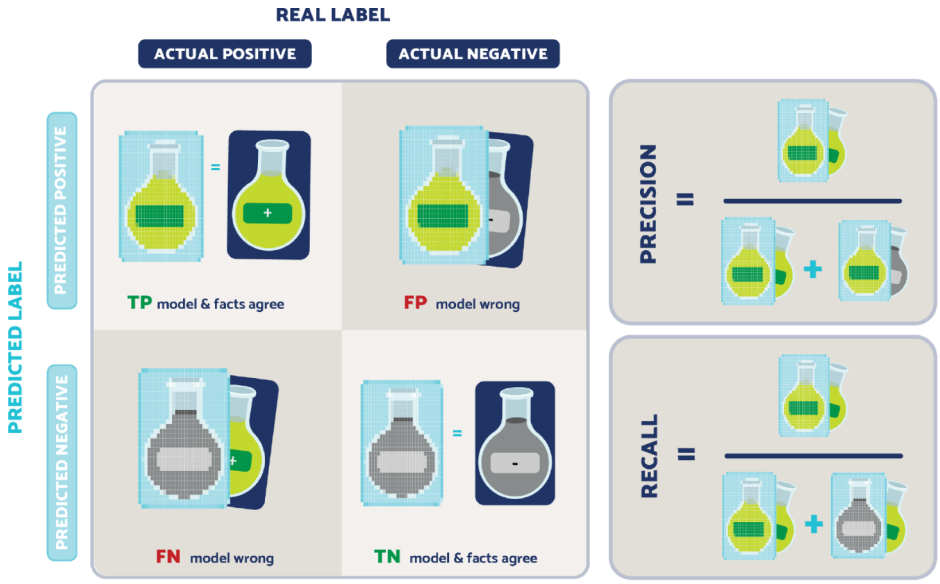


Figure 1. Illustration of evaluation metrics—the blurred symbol is the model's prediction, and the unblurred symbol is the true label of the data.

choice of the training and test subsets, making it difficult to obtain a reliable estimate of the model's generalization error. In such cases, providing an accurate and unbiased estimate of the error through cross-validation (CV) and hyper-parameter tuning becomes essential, which in turn allows for the selection of the most robust and best-performing model.[21,43]

In our study, we use a variant of nested *k*-fold cross-validation to reliably evaluate model performance on unseen data.[44,45] In *k*-fold cross-validation, the data set is divided into k-subsets of roughly equal size, one of which is chosen as the validation set, another one as the test set, and the rest are combined into the training set. The model is then trained using the training set, while the validation set is used to select the best-performing set of hyperparameters during training. Finally, the test set is used to evaluate the model's predictive power on unseen data. This procedure is then performed for different allocations of the subsets to the training, validation, and test data sets. Nested *k*-fold cross-validation improves robustness by creating multiple different random splits of the data set into k subsets, and performing the whole process of *k*-fold cross-validation multiple times. Using this procedure, we ensure that each data point is represented in the train, validation, and test set in different splits, preventing overfitting and ensuring unbiased performance evaluation that is not dependent on the initial partitioning of the data.

Machine Learning Models. To showcase the general nature of our framework and provide a broad overview of the diverse approaches in machine learning, we evaluate a variety of ML models commonly used in classification tasks.

This includes a series of models from the family of tree-based models such as *decision trees* (with both prepruning and postpruning), as well as *random forests* and *gradient boosted trees*, which are ensembles constructed of many individual decision trees. We also include logistic regression, one of the oldest and most popular methods for binary classification.

Finally, we evaluate SVMs and neural networks, two powerful and highly nonlinear ML methods. Detailed explanations of these models can be found in Supplementary Section Machine learning models theory.

Explainable AI. Explainable AI (XAI) techniques are playing an increasingly important role in various domains, including catalyst research.[46] While there are many approaches to explaining the ML model's decisions, in this paper, we focus on XAI methods that assign importance to each input feature based on how relevant they were to the model's prediction.[27] In the catalyst design scenario explored here, such XAI methods would point out which components contribute particularly strongly to a catalyst being classified as either high- or low-yield. Considering the black-box nature of machine learning models, by implementing XAI techniques, we can discern whether the model is focusing on chemically relevant features rather than artifacts. This helps to prevent issues like the "Clever Hans" effect,[24] enhancing the model's transparency and interpretability. In the ideal case, these techniques can also uncover previously unknown relationships between catalyst components, thereby guiding the exploration of new catalysts and advancing the field.[47] However, we would once again like to note, that the main goal of XAI methods is to bring better transparency and understanding of the decision making of ML models, rather than uncovering new knowledge.

Feature Importance for Tree-Based Models. For the three variants of decision trees, feature importance was determined based on the mean decrease in impurity across all decision nodes. This metric quantifies the contribution of each input feature to reducing impurity when splitting the data along this feature during the training process,[48] commonly measured by the Gini index or entropy.

$$\mathrm{Gini}(t) = 1 - \sum_{i=1}^{J} p_i^2 \tag{1}$$

where J is the total number of classes in the data set, and p_i represents the proportion of samples belonging to class i at node t.

In this case, since the decision tree models were trained using the Gini index, we also use this as the measure of feature importance. Higher importance scores indicate a greater impact on impurity reduction, highlighting the significance of these features in the classification process. For the random forest models, feature importance is determined by aggregating

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

the reduction in Gini impurity achieved by splitting each feature across all trees within the ensemble:

$$FI_{RF}(\mathbf{x}_d) = \frac{1}{T}\sum_{t}^{T}\sum_{s}^{S_t}\Delta\text{Impurity}_{t,s}(\mathbf{x}_d) \tag{2}$$

Here, $\mathbf{x}_d$ is the d-th feature of the input vector $\mathbf{x}$, T is the total number of trees in the random forest, S_t is the number of splits in tree t, and $\Delta\text{Impurity}_{t,s}(\mathbf{x}_d)$ is the decrease in Gini impurity attributable after split s in tree t, if feature $\mathbf{x}_d$ was used.[49]

In eXtreme Gradient Boosting (XGBoost), a feature's importance increases with its contribution to splits during tree construction and is calculated by summing the gain (see Supplementary Section XGBoost models) of each specific feature across all trees and splits:

$$FI_{XGB}(\mathbf{x}_d) = \sum_{s}^{S}\text{Gain}_s(\mathbf{x}_d) \tag{3}$$

where $\mathbf{x}_d$ once again refers to the feature d in the input $\mathbf{x}$, S is the total number of splits across all trees, and $\text{Gain}_s(\mathbf{x}_d)$ is the gain resulting after split s, if feature $\mathbf{x}_d$ was used for this split.

Layer-Wise Relevance Propagation (LRP). LRP is a popular explaining technique for interpreting predictions of complex neural network models in terms of latent and input features.[25,26,28] In contrast to feature importances for tree-based models, which primarily explain the parameters of the model itself, LRP produces local explanations for the classification of each sample. Using so-called propagation rules,[50] LRP assigns a relevance value to each neuron by iteratively backpropagating the model output through the network layers until the input layer is reached. Propagation rules are chosen to be conservative, meaning that total relevance in each layer is equivalent to the network output. In general, most LRP rules compute lower-layer relevance R_i given upper-layer relevance R_j using the following generic format:

$$R_i = \sum_j \frac{\rho(w_{ij})\cdot a_i}{\sum_{0,i'}\rho(w_{i'j})\cdot a_{i'} + \epsilon}\cdot R_j \tag{4}$$

In the above formulation, the sum $\sum_j$ runs over upper-layer neurons $\{a_j\}_j$, whereas the sum $\sum_{0,i'}$ runs over lower-layer neurons $\{a_i'\}_{i'}$ including the bias represented as the additional neuron a_0. The variable w_{ij} describes the weight connecting the lower-layer neuron activation a_i and the upper-layer neuron a_j, while ρ describes some functional dependence of the neuron weights. To avoid division by zero, most LRP rules stabilize the above denominator adding a small positive value ϵ. As exemplified by the above formula, most propagation rules distribute relevance depending on how much each lower-layer neuron has contributed to the output of the higher-layer neuron. Contrary to feature importance explanations, LRP relevance values can be either positive or negative, thus describing how much a given feature attributed to the model deciding in favor of one class or the other.

To start relevance propagation, a suitable neuron output must be chosen to set upper-layer relevance. One possible set of explained neurons is the neurons in front of the final softmax layer, which aggregate evidence for a given class. The upper-layer evidence neurons form a linear layer and compute activations for a given class c as follows:

$$a_c = \sum_{0,k} w_{c,k}\cdot a_k \tag{5}$$

However, as it has been found that explaining only one class-evidence neuron does not contextualize evidence of competing classes, an alternate approach is to explain the logit of class probabilities instead.[50] This quantity is expressed as follows:

$$\eta_c = \frac{\log(p_c)}{\log(1-p_c)} \tag{6}$$

In a two-class setting with class indices 1 and −1, this further simplifies as follows:

$$\eta_1 = a_1 - a_{-1} \tag{7}$$

Combining the evidence weight vectors, η can then be expressed as the following explainable neuron:

$$\eta_1 = \sum_{0,k}(w_{1,k} - w_{-1,k})\cdot a_k \tag{8}$$

This neuron η can then finally be used as the starting point of the relevance propagation procedure for the classifier. To explain subsequent Multi-Layer Perceptron (MLP) layers, we applied the γ-rule, which sets the functional dependence $\rho(w_{ij}) = w_{ij} + \gamma\cdot\max(0, w_{ij})$ given a value of γ, that we set to 0.2. The γ-rule emphasizes positive contributions to neuron outputs, which has been shown to improve the stability and faithfulness of the resulting explanation.[50]

While the LRP rules conserve relevance, some relevance in each layer gets assigned to neuron bias terms, which cannot be explained in terms of input features. Consequently, the relevance assigned to the input does not perfectly match the class evidence of the prediction. To account for the relevance lost to the biases and improve the interpretability of the relevances assigned to the input, we rescale the input feature relevance values $\{R_d\}_d$ such that positive relevance adds up to the positive-class evidence and vice versa.

The rescaling is done using sign-dependent factors ρ^+ and ρ^-, which are determined based on the positive and negative contributions to the output neuron η_1. More specifically, for each sample, the positive input relevances are scaled such that their sum is equal to the positive contributions to η_1, and vice versa for the negative relevance:

$$\sum_d \rho^+\cdot\max(R_d, 0) = \sum_{0,k}\max((w_{1,k} - w_{-1,k})\cdot a_k, 0) \tag{9}$$

$$\sum_d \rho^-\cdot\min(R_d, 0) = \sum_{0,k}\min((w_{1,k} - w_{-1,k})\cdot a_k, 0) \tag{10}$$

This rescaling strategy ensures that the relevance in the inputs conserves the value of the output neuron η_1 in a way that preserves the original sign of the input relevances.

LRP for Neuralized SVMs. By default, LRP requires a neural network structure and is, without further modification, not suited to explain kernel-based models. To overcome this limitation and provide faithful explanations with LRP, Kauffmann et al.[51] introduced the concept of neuralization. Neuralization transforms a kernel-based model into a neural network structure producing equivalent decisions explainable with propagation-based XAI methods.

In the case of RBF-SVMs, Bley et al.[52] modified the SVM predictive function $f(\mathbf{x})$ of eq 10 in the following way:

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

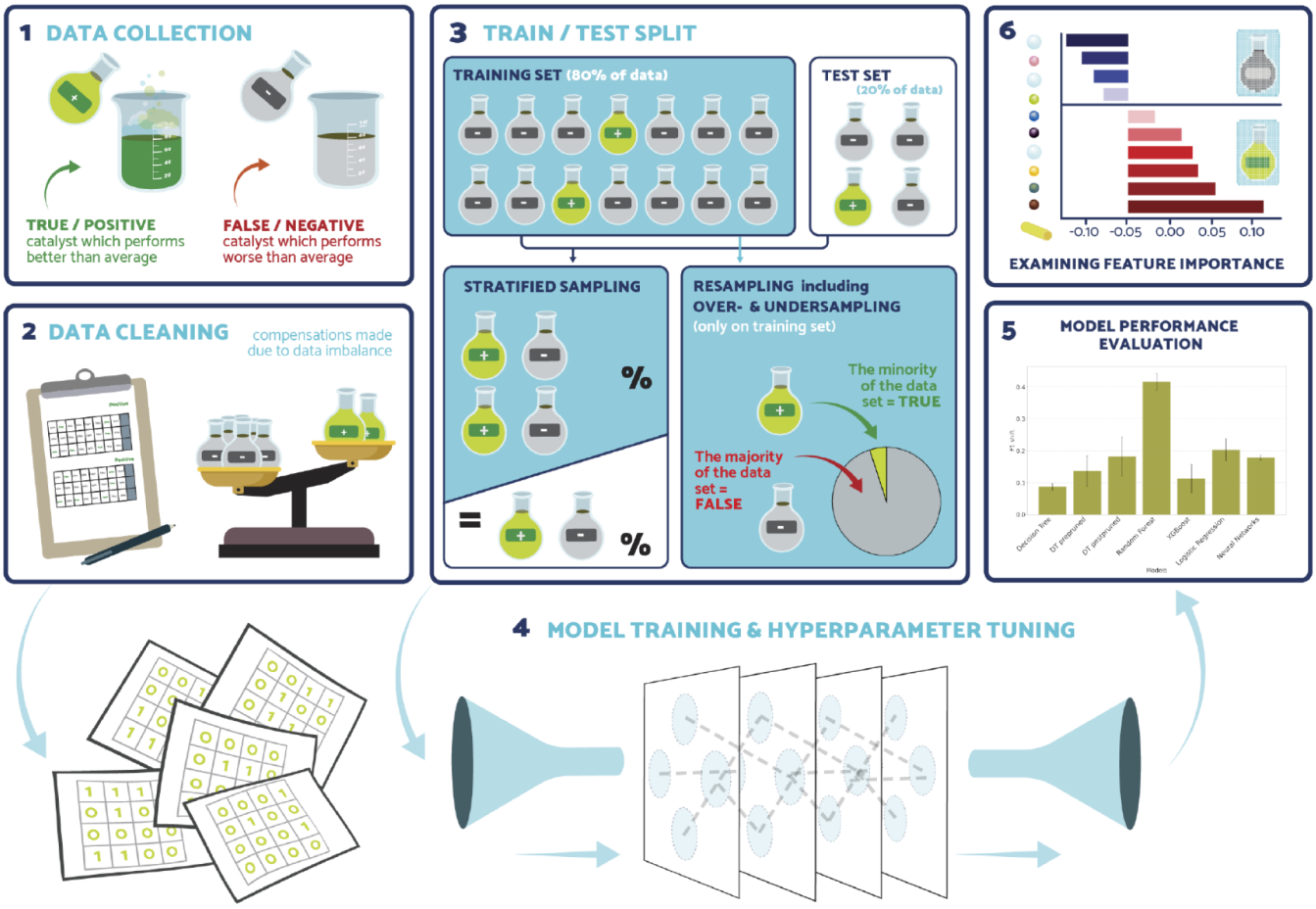


Figure 2. Illustration of the ML framework, starting with data collection and cleaning (steps 1−2), and visualizing the process for obtaining the performance and explanations of a model on a single random train-test split of the data set (steps 3−6). These training and evaluation steps are then repeated for 100 different train-test splits, and the results are aggregated to produce robust performance estimates and feature importance scores.

$$g(\mathbf{x}) = \log\left(\sum_i \alpha_i \exp(-\gamma\cdot\|\mathbf{x}-\mathbf{x}_i\|^2)\right) - \log\left(\sum_j |\alpha_j| \exp(-\gamma\cdot\left\|\mathbf{x}-\mathbf{x}_j\right\|^2)\right) \tag{11}$$

Here, the index i runs over positive-class support vectors and j over the negative-class support vectors. Therefore, $\mathbf{x}_i$ and $\mathbf{x}_j$ describe the support vectors themselves, and α_i and α_j are the associated dual coefficients of the SVM. The two logarithmic terms can be interpreted as evidence for the two competing classes. The transformed classifier $g(\mathbf{x})$ is guaranteed to produce an equivalent classification to the original SVM. The authors went on and transformed $g(\mathbf{x})$ into the following neural network structure:

$$\begin{aligned} g(\mathbf{x}) &= \gamma\cdot\min_j{}^{\gamma}(\max_i{}^{\gamma}(w_{ij}^T\cdot\mathbf{x}+b_{ij})) \\ \text{where } w_{ij} &= 2\cdot(\mathbf{x}_i-\mathbf{x}_j), \\ b_{ij} &= \left\|\mathbf{x}_j\right\|^2 - \left\|\mathbf{x}_i\right\|^2 + \frac{1}{\gamma}\cdot\log\left(\frac{\alpha_i}{|\alpha_j|}\right) \end{aligned} \tag{12}$$

The above formulation utilizes the soft-min and soft-max pooling-layer definitions of Kauffmann et al.[51] where $\min^{\gamma}(\cdot)$ is defined as $-\frac{1}{\gamma}\log\sum\exp(-\gamma\cdot(\cdot))$, and $\max^{\gamma}(\cdot)$ is defined as $\frac{1}{\gamma}\log\sum\exp(\gamma\cdot(\cdot))$. Thus, the neuralized RBF-SVM can be summarized as two pooling layers preceded by one detection layer with one detection neuron for each pair ij of positive-class and negative-class support vectors.

To propagate relevance through the first two pooling layers, we follow the approach of Kauffmann et al.[51] Based on the concept of Deep Taylor Decomposition,[50] the authors derived the following conservative propagation rules for the soft-min and soft-max layers:

$$R_j = \frac{\exp(-a_j)}{\sum_{j'}\exp(-a_{j'})}\cdot R_k \tag{13}$$

$$R_{ij} = \frac{\exp(a_{ij})}{\sum_{i'}\exp(a_{i'j})}\cdot R_j \tag{14}$$

To propagate through the linear layer and produce input feature relevance values $\{R_d\}_d$, we use the LRP-0 rule, which attributes relevance according to the element-wise product of the neuron weights and input features:

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

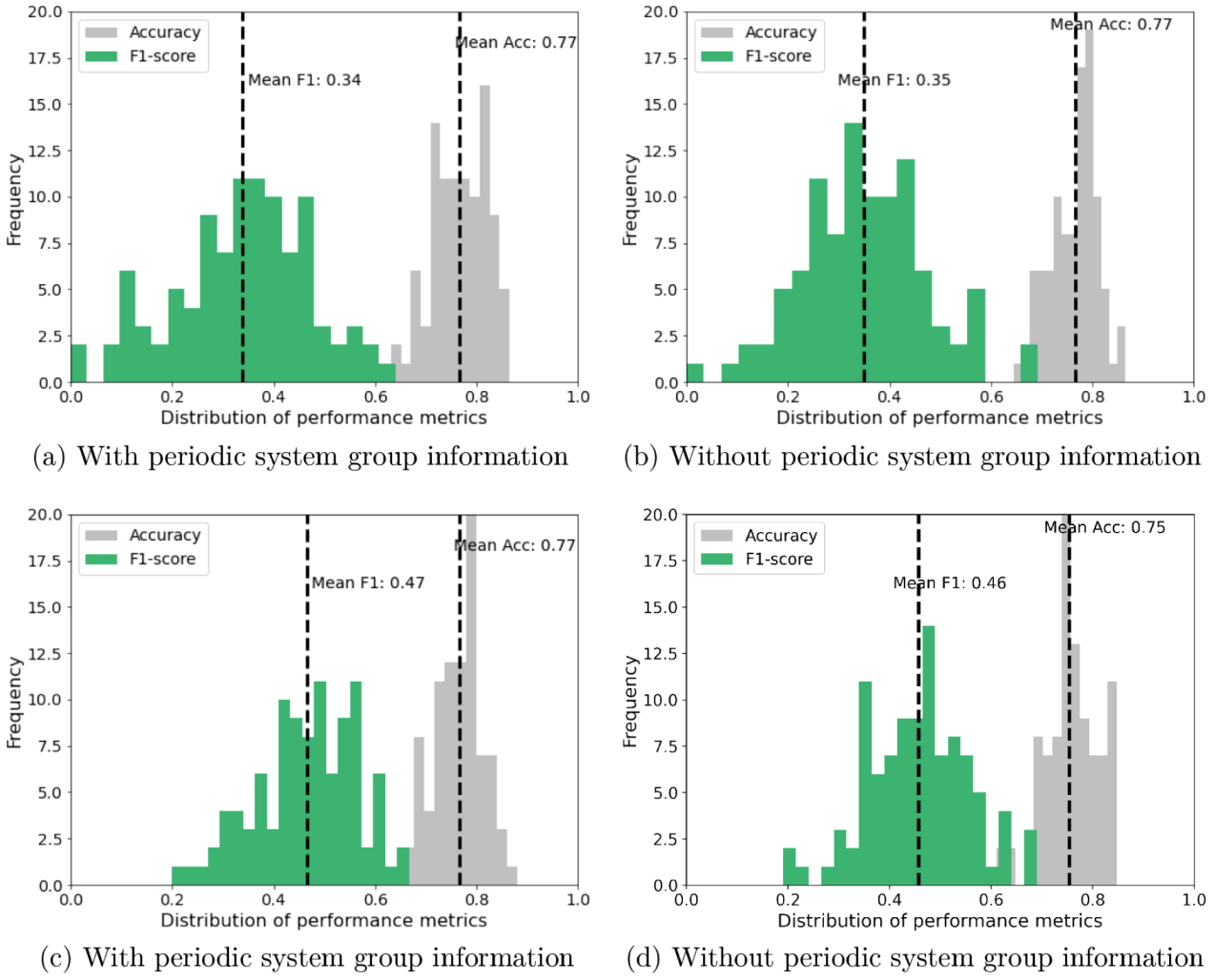


Figure 3. Comparative analysis of accuracy and F1-score distribution for a decision tree over 100 evaluation cycles. Figures (a) and (b) show results without our ML framework, while Figures (c) and (d) present results with our ML framework. The vertical line shows the mean of the respective distribution.

$$R_d = \frac{w_{ij,d}\cdot \mathbf{x}_d}{\sum_{0,d'} w_{ij,d'}\cdot \mathbf{x}_{d'}}\cdot R_{ij} \tag{15}$$

During this propagation, however, relevance is naturally lost in the linear layer to the biases. To compensate for lost relevance and ensure interpretability, we reweight relevance such that positive and negative relevance add up to the original class evidence of the explained model in eq 11. In particular, we identify sign-dependent reweighting factors ρ^+ and ρ^- to rescale feature relevance such that the following relations hold:

$$\sum_d \rho^+\cdot\max(R_d, 0) = \log\left(\sum_i \alpha_i \exp(-\gamma\cdot\|\mathbf{x} - \mathbf{x}_i\|^2)\right) \tag{16}$$

$$\sum_d \rho^-\cdot\min(R_d, 0) = -\log\left(\sum_j \alpha_j \exp(-\gamma\cdot\left\|\mathbf{x} - \mathbf{x}_j\right\|^2)\right) \tag{17}$$

RESULTS AND DISCUSSION

To ensure the reliability and best practices for evaluating ML methods in catalyst design, we elaborate on our proposed ML framework tailored for data sets characterized by small-scale and class imbalances. This section provides an overview of the framework's architecture and its application to various ML models. We then assess the impact of different framework components (e.g., performance measures, resampling techniques) on model performance. Finally, we leverage XAI techniques to analyze the most relevant features identified by each model and investigate common features across models to gain an understanding of the underlying data. All of the following experiments were conducted on a MacBook Pro with an M1 chip and 36 GB of RAM. The code for our framework and the experiments in this section is available at https://github.com/PSemnani/XAI4CatalyticYield.

Evaluation Framework. In this section, we will describe the ML framework tailored to address the challenges posed by limited and unbalanced data, illustrated on Figure 2.

We begin with data acquisition, followed by data cleaning and preprocessing steps to refine the data set for further analysis. During the training process, we use stratified sampling to ensure equal representation of all classes across the training and test sets. This is followed by targeted resampling within the training set, addressing the significant imbalance within our data set of 291 samples, where only 51 are positive. Considering the 80−20 train-test split, this leads to about

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

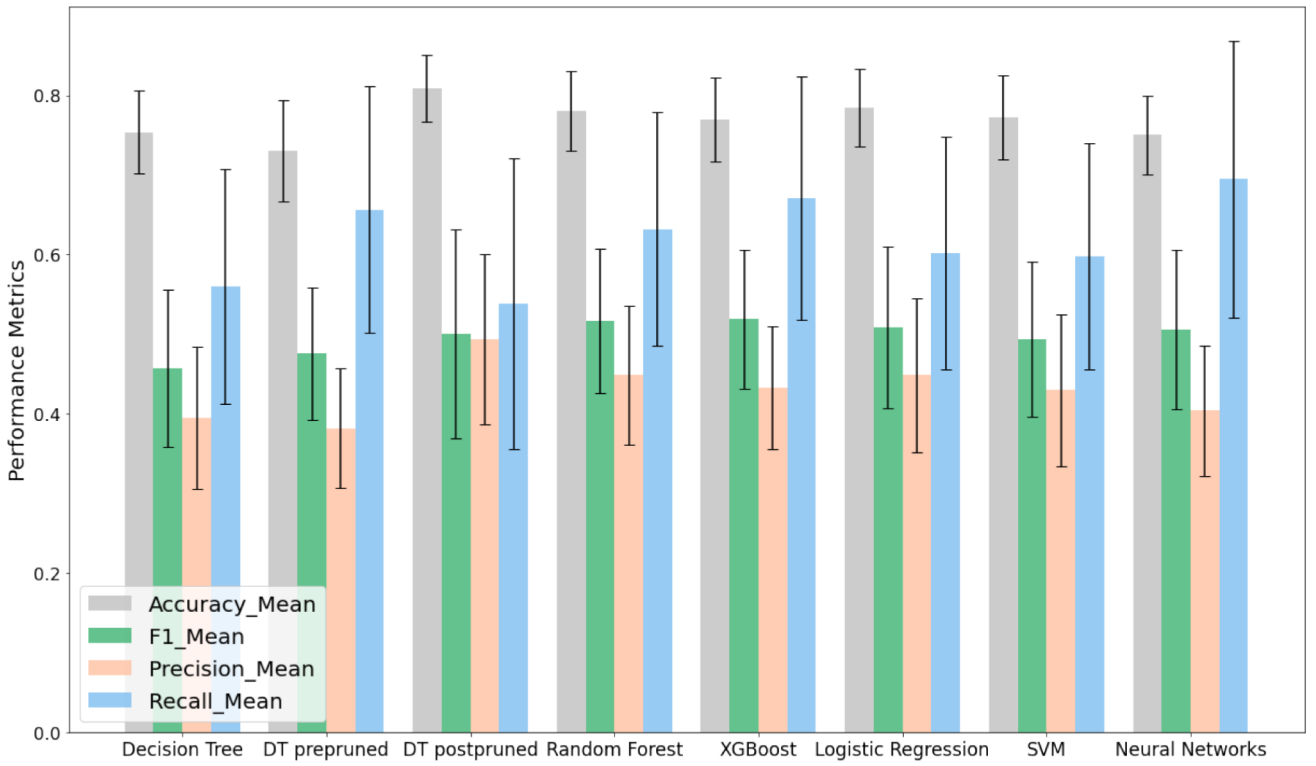


Figure 4. Model performance evaluation with four different evaluation metrics, for all of the discussed ML models. The bars demonstrate the mean of the respective metric, and the error bars present their standard deviation.

230 training samples, with approximately 41 positives initially. In order to balance the data set for training, we apply SMOTE[42] with an oversampling ratio of 0.6 for the minority class. This increases the number of positive samples in the training set after resampling to approximately 60% of the majority class, resulting in about 115 positives. The following undersampling (ratio 1) maintains the total number of samples intact. SMOTE generates new and unique samples by mixing neighboring samples of the minority class, ensuring each is slightly altered and distinct.

The model training process incorporates k-fold cross-validation with $k = 5$ for hyperparameter tuning, designed to improve the predictive accuracy and generalizability of our machine learning models. Hyperparameter tuning is performed with Bayesian optimization using Gaussian processes, where the optimization is conducted over a range of reasonable parameters, which are detailed in the Supplementary Section ML models training and feature importance scores results. This comprehensive approach ensures a robust evaluation of model performance across diverse parameter settings. Since we chose a more constrained set of hyperparameters for the neural network models, we do a grid search across all combinations of hyperparameters instead of Bayesian optimization. We evaluate each model's performance based on accuracy and F1-score metrics. Furthermore, we assess the importance of different features in the data set via XAI techniques.

To mitigate potential biases due to limited data (291 data points) and variability in the train-test splits, we perform the random splitting of the train and test set and subsequent steps of resampling and evaluation 100 times (steps 3 to 6). The cross-validation is then performed by splitting the training set into subsets. This ensures that the models are trained and evaluated with no information leakage from the test samples. The process results in a nested k-fold cross-validation, providing reliable model performance estimates through averaging. This ensures better generalizability of our results under small, imbalanced data sets.

Robust Performance Estimation. In Nguyen et al.,[16] a decision tree model was created using a single train-test split. However, when we tried to replicate this model using an alternative split, we found the model's performance scores were highly inconsistent. This inconsistency is demonstrated by the wide range of accuracy scores across 100 different splits and random states, as shown in Figure 3a. The accuracy score of the single decision tree model in Nguyen et al.[16] was 0.78, which is very close to the mean of the distribution in the figure, with a value of 0.77.

These variations can be attributed to several factors. First, the randomness inherent in data-splitting results in different subsets being used for training and testing. The sensitivity of decision trees to the training data distribution can, therefore, create inconsistencies in model performance due to these variations in the training set, which are especially high for smaller data sets. In addition, decision tree algorithms often incorporate random initialization of parameters such as feature selection and node splitting thresholds, resulting in different trees being generated at each training iteration, further amplifying the variance of the model's performance.

When dealing with imbalanced data sets, we argue strongly against the reliance on accuracy as the primary performance metric for ML classifiers, due to the susceptibility to misinterpretation of the accuracy scores. When one class significantly outweighs the others, accuracy tends to be skewed, favoring models that simply predict the dominant class. This phenomenon is evident in our data set, where among 291 data points, only 51 are labeled as positive

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

catalysts. Consequently, if a model consistently labels catalysts as negative, its accuracy would approximate the frequency of the dominant class, yielding a high but misleading accuracy score of 0.82. Our aim in catalyst material discovery extends beyond recognizing prevalent classes to accurately predicting out-of-distribution samples or classes with fewer samples. The F1-score, by considering both precision and recall and focusing mainly on the positive class, offers a better estimate of a model's performance in imbalanced catalyst design scenarios.

An interesting observation emerges from the comparison between accuracy and F1-score in Figure 3a: while accuracy appears to be satisfactory, F1-score shows a much wider range with lower values. This gap suggests a notable weakness in the model's predictive capacity, particularly for high-yield catalysts.

By using more stable data splitting methods such as stratified sampling and ensuring class balance via resampling strategies, our proposed framework aims to reduce variability and thus improve the reliability of the estimated performance of decision tree-based predictive models. This effect is supported by the fact that the new framework produced a narrower spread of performance metrics, as seen in Figure 3c, indicating a more consistent and robust training process.

The analysis so far has included the features from the periodic system groups introduced in Nguyen et al.[16] in order to make the comparison fair. However, as we mentioned in Subsection Data, we found that these group features do not contribute to the performance of the model. As seen in Figure 3c,d, the exclusion of the periodic table group features does not result in any significant change in the distribution of performance scores. Because of this and the issue of explainability as outlined in Supplementary Section Effect of periodic table group features, from here on, we will only report results using the data set without the periodic table group features, i.e., using only information about which elements and supports were present in the catalyst.

Evaluating Other Machine Learning Models. Based on the findings of the previous section, the appropriateness of the decision tree model for our data set comes under question, prompting us to explore alternative modeling approaches. We first turn our attention to other tree-based techniques such as pre- and postpruned decision trees, random forest, and XGBoost. To cover a more diverse range of ML approaches, we extended our analysis to nontree models such as logistic regression, SVMs, and neural networks.

Performance Evaluation. Following the suggested framework, the performance metrics of all models are calculated and displayed in Figure 4 and Table 2. The mean value of accuracy across various models lies between the small range of 0.73 to 0.81. The best-performing model is the postpruned decision tree, with an accuracy of 0.81. However, as we discussed in the previous section, due to the imbalanced ratio of both classes in the OCM data set, a model classifying all catalyst samples as only having negative performance would have an accuracy of 0.82, demonstrating how misleading using accuracy is as a measure of performance in this case.

A look at the F1-score on the other hand, which takes the under-representation of the high-yield catalysts into account, paints a different picture. For reference, given the data set's class ratio, the F1-score of a random classifier would be 0.26, a classifier predicting only negative performing catalysts would have an F1-score of 0.0, while a classifier predicting only the positive class would yield an F1-score of 0.3. With this in mind, the results in Table 2 demonstrate that all the models have performed significantly better than the random classifier, with F1-scores ranging from 0.46 to 0.52. Given that this difference in performance was impossible to recognize based on the accuracy, we can conclude that in the context of imbalanced classes, the F1-score is much more informative as a measure of the model's performance compared to accuracy.

Table 2. Model Performance Evaluation Results Implemented through the Suggested ML Framework[a]

Model	Accuracy Mean	Accuracy Std	F1 Mean	F1 Std
Decision Tree	0.75	0.05	0.46	0.10
Decision Tree Prepruned	0.73	0.06	0.47	0.08
Decision Tree Postpruned	0.81	0.04	0.50	0.13
Random Forest	0.78	0.05	**0.52**	0.09
XGBoost	0.77	0.05	0.51	0.09
Logistic Regression	0.78	0.05	0.51	0.10
SVM	0.77	0.05	0.49	0.09
Neural Networks	0.76	0.05	0.51	0.10

[a]The accuracy and F1-score of each model are averaged over 100 training and test splits and compared, and their respective mean and standard deviation are displayed here.

It is important to emphasize that while the reported F1-score range is significantly higher than that of a random classifier and provides more information than accuracy, the various machine learning models show fairly comparable performance, with none distinctly outperforming the others. This can be mainly attributed to the small size of the data set used for training. As a result, the models exhibiting the best performance—after thorough cross-validation and hyperparameter tuning—are generally simpler in terms of their parameters and complexity. This simplicity leads to more robust predictions and minimizes the risk of overfitting, allowing the models to achieve satisfactory validation and test errors. Consequently, the models demonstrate similar quality given the current data set conditions. These results reflect the data set's characteristics rather than the models' inherent capabilities. With a larger and higher-quality experimental data set, we expect to see more significant performance differences, especially among more complex models like XGBoost, SVM, and neural networks, which are likely to outperform simpler models such as decision trees, random forests, and linear regression.

Impact of Resampling. To emphasize the benefits of our suggested ML framework, we performed the same evaluation procedure without the resampling step when preparing the training data. Figure 5 illustrates the impact that resampling has on the different performance metrics across all models. We observe a minor drop in accuracy for all models, however, this is of little importance since we already determined that accuracy is not an appropriate performance measure in this context.

On the other hand, the application of resampling during training has significantly improved the F1-score for all models besides the SVM. The SVM in general is not strongly affected by class imbalance since it relies only on a few samples as support vectors, which usually lie on the edges of the class distributions. A resampling method such as SMOTE that generates artificial samples by mixing existing data points of the same class is thus unlikely to generate any new samples on the edge of the distribution. On the other hand, the random forest has benefited the most from the introduction of resampling, with its F1-score increasing from 0.1 to 0.52.

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

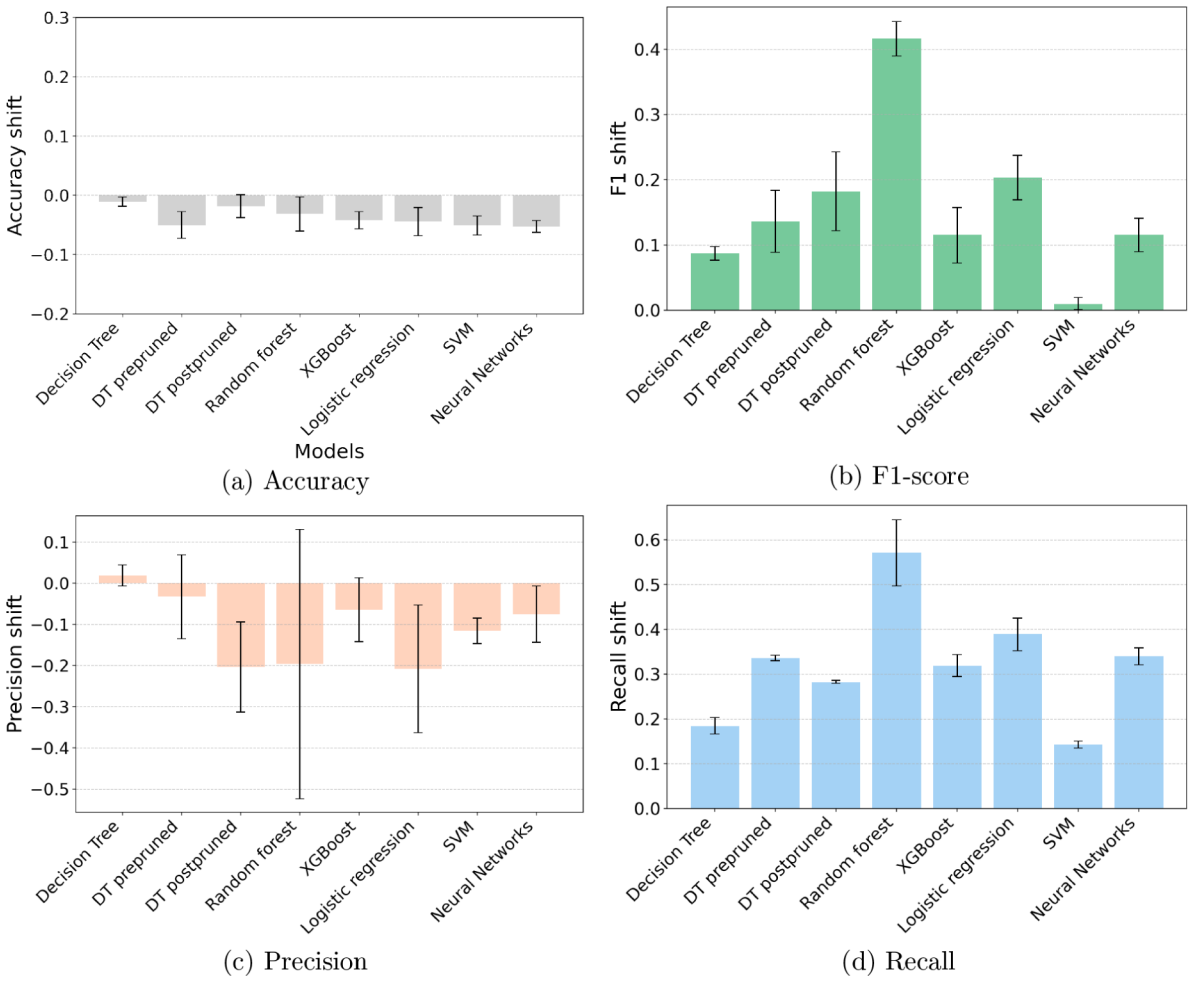


Figure 5. Impact of introducing resampling techniques on key performance metrics: (a) accuracy, (b) F1-score, (c) precision, and (d) recall.

The overall improvement in F1-score can be attributed to a significant increase in recall across all models. This indicates that resampling enables the models to better identify the minority class of high-yield catalysts, which is the primary class of interest in catalyst design. We also observe a small reduction in precision for most of the models, revealing that the proportion of false positives has slightly increased as a consequence of the models classifying more catalyst compositions as high-yield.

Given the substantial improvements in recall and F1-score, we can confidently conclude that our machine learning framework effectively enhances model performance and reliability for catalyst yield classification.

Explaining the Decisions of ML Models. Despite the challenges observed in accurately predicting catalyst yield, ML models offer more than just predicting accuracy; they can serve as valuable tools for analysis. In this section, we use the previously trained ML models to explore the underlying factors that drive catalyst performance. For this purpose, we apply a range of XAI methods to identify the most influential features for classifying a sample as "good" or "bad". For each model class, we conducted an aggregation procedure as described in Subsection Evaluation Framework: For each of the 100 training-test splits, cross-validation was used to identify optimal hyperparameters. These hyperparameters were then used to train a single model on the combined train and validation set for each specific split. The test data set was subsequently used to estimate the model's generalization performance and generate sample-specific explanations, if necessary. The relevances assigned to all data points were averaged over the number of splits to reduce the effect of model's bias due to specific subsets chosen for training. These aggregated results help identify common patterns and key contributors to catalyst performance, which can provide chemists with insights that can guide future experimental strategies.

Feature Importance for Tree-Based Models. In order to aggregate the importance score of features across all tree-based models, we have first normalized these values between zero to one and then took the mean of the importance score for each feature:

$$\overline{R}_d = \frac{1}{S}\sum_{i=0}^{S} R_d(m^{(i)}) \tag{18}$$

where S is the number of training/test splits, $R_d(m^{(i)})$ is the feature importance for feature d extracted from the model m trained on the training subset from split i. The results of the feature importance aggregation are illustrated in Figure 6. Manganese (Mn) was found to be the key feature in catalyst yield prediction, followed by Aluminum Oxide (Al_2O_3), Silicon Dioxide (SiO_2), Nickel (Ni) and Cerium Dioxide (CeO_2).

Overall, the feature importance scores assigned across the different tree-based models are very similar. As shown in Figure 10, the correlation coefficients of the importance scores

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

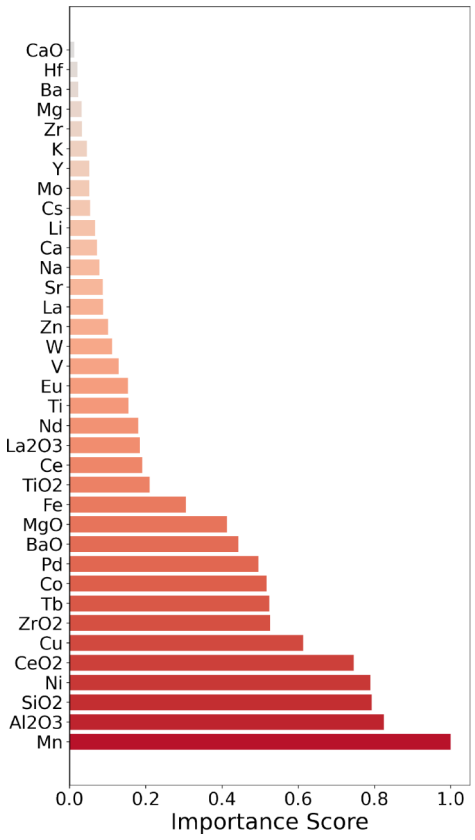


Figure 6. Averaged importance scores for all features across the different tree-based models (decision tree, DT prepruned, DT postpruned, random forest, XGBoost).

between the tree-based models are all over 0.94. Some part of this high similarity of the feature importances may be explained by the similarity of the explanation methods themselves since the explanations of tree-based models are all based on the reduction of impurity related to each feature. Another and perhaps more significant reason for the similar explanations of the tree-based models is that they all fundamentally use a similar learning strategy of selecting features that reduce the impurity in the leaf/decision nodes.

Explanations Using LRP. To produce explanations for catalyst yield in SVMs, we performed the neuralization procedure outlined in Section LRP for neuralized SVMs and applied the propagation rules to obtain relevance scores for each input feature of each test sample. To counteract relevance lost to bias terms, we rescaled input relevance using our rebalancing scheme described in eq 16.

Similarly to the SVM, neural network explanations are obtained by applying LRP as outlined in Section Layer-wise Relevance Propagation (LRP) to each test sample. Again, in order to correct the relevance loss because of the model's bias parameters and maintain the conservation of relevance between the input and output, we rescale the input relevance as shown in eq 9.

LRP is an explanation method that inherently produces individual explanations for each sample (for some examples of single sample LRP explanations, see Supplementary Section Single catalyst explanation with LRP). Therefore, to obtain global feature importances based on the entire data set, it is insufficient to aggregate across the different training/test splits as in eq 18, the sample-based explanations within each split also need to be aggregated. This aggregation procedure remains the same for the LRP explanations of both the SVM and neural network models. The rescaled feature relevances for all test samples are averaged across each sample from each of the 100 test splits:

$$\overline{R}_d = \frac{1}{S * N} \sum_{i=0}^{S} \sum_{j=0}^{N} R_d\left(\mathbf{x}_j^{(i)}\right) \tag{19}$$

where S is the number of training/test splits, N is the number of test samples per split, $R_d(\mathbf{x}_i^{(j)})$ is the relevance for input feature d of the j-th sample in the test subset for split i.

We stress that LRP explanations yield both positive and negative values, unlike tree-based feature importances, which only produce positive relevance values. Due to our choice of the evidence for the high-yield class as a starting point for the LRP propagation, a positive relevance at the input indicates that this feature contributes positively to the model's prediction of the high-yield class, while the features with negative relevance contribute toward the model classifying the catalyst as low-yield. In contrast, tree-based feature importances only indicate a feature's overall importance without specifying its relation to a particular class.

While aggregated LRP explanations using signed importance scores provide more nuanced information about model behavior, they are not directly comparable to strictly positive tree-based explanations. To enable a direct comparison, we also aggregate absolute LRP relevances across the different samples and splits, providing purely positive feature importances.

The resulting aggregated signed and absolute feature importances for the SVM model can be seen in Figure 8, while the analogous visualizations of the average feature importances for the neural network model are shown in Figure 7.

For the neural network models, the highest absolute relevance scores have been assigned to Nickel (Ni) and Manganese (Mn) alongside alumina (Al_2O_3) and silica (SiO_2). These components are, therefore, key features for the classification of a catalyst as either high- or low-yield, according to the neural network. Mn and Al_2O_3 have also been identified as top features by SVM models. However, they are preceded by the supports La_2O_3, BaO, which have been assigned even higher importance scores.

Thanks to the property of LRP to assign positive and negative relevances to features, the signed averaged LRP importances provide a further dimension for analysis compared to the absolute feature importances. We observe that all of the top absolute contributors identified by both neural networks and SVM have been the highest "negative" contributors to classifying a catalyst as "high-yield", namely Ni, Mn, and Al_2O_3, while La_2O_3, BaO and Eu are the key components for classifying a catalyst as high-yield according to both the neural network and SVM models.

Similarity of Explanations. The first thing to note about the absolute feature importances across the tree-based models, neural networks, and SVMs (Figures 6, 7a, and 8a) is that Manganese (Mn) has been identified as one of the most critical elements for determining the yield of a catalyst, accompanied by the support material alumina (Al_2O_3), which also has universally high importance. Both of these components are the only ones to appear among the top 5 components in terms of absolute relevance across all three types of models.

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

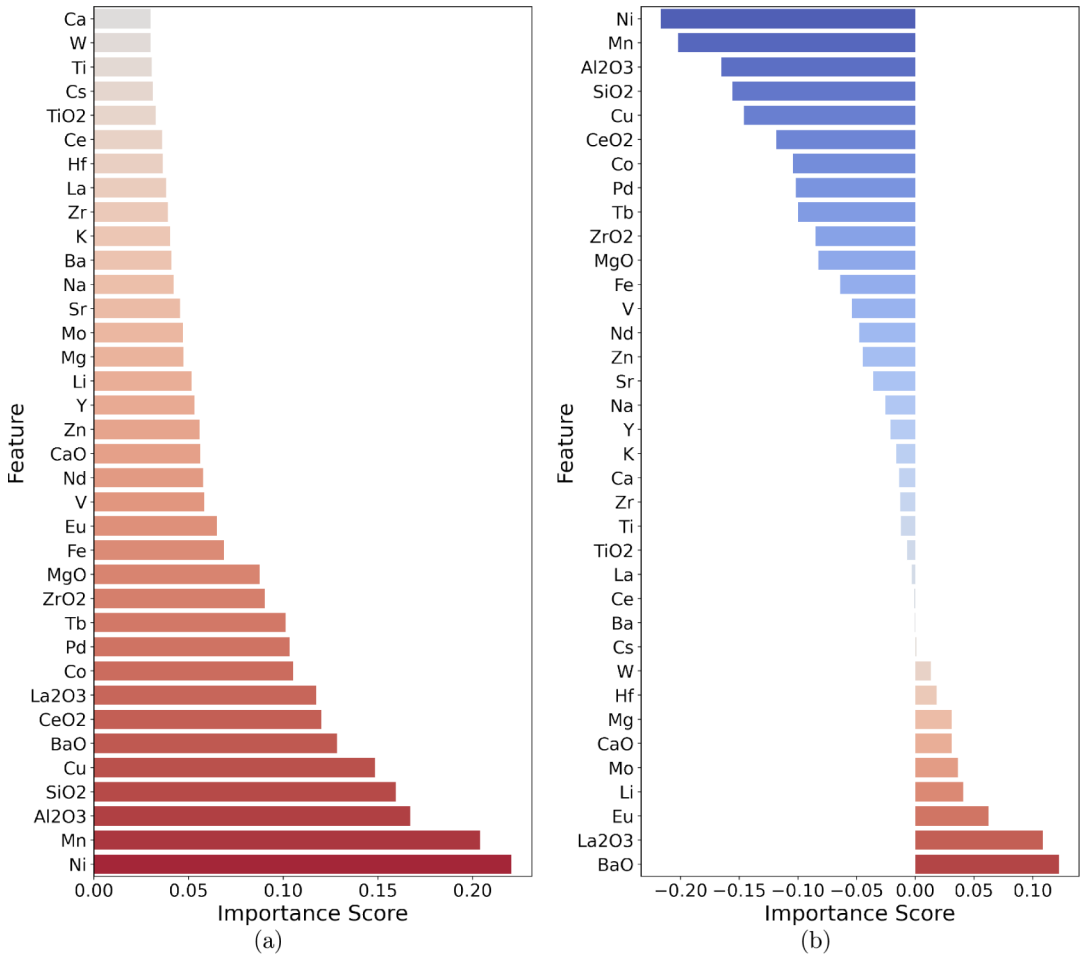


Figure 7. Mean of feature importance analysis for neural networks via LRP based on the classifier score of the high-yield class. (a) Mean absolute feature importances to identify the key features independently of class-specific relevance. (b) Mean feature relevance, including positive and negative relevances, disentangling the class-specific contributions of the inputs.

Figure 9 visualizes the average absolute feature importance as well the standard deviation across ML models of different types: SVM, neural networks, logistic regression, and random forest, as a representative of the tree-based models. We find that the top three key metals in determining the yield of a catalyst are Manganese (Mn), Nickel (Ni), and Copper (Cu) and the top three support materials are alumina (Al_2O_3), silica (SiO_2) and cerium dioxide (Ce_2O). Lanthana (La_2O_3) in particular stands out as having a high standard deviation, owed to the fact that logistic regression and tree based models assign very low feature importance to La_2O_3, while it is absolute feature importance for neural networks and especially SVMs is much higher, because this two models specifically identify La_2O_3 as highly relevant to high-yield catalyst compositions.

Furthermore, we performed an analysis aimed at identifying similarities and distinctions between the different models in terms of feature importance. This was achieved by calculating the Pearson correlation coefficients between the feature importance scores of each pair of models via the Fisher-Z transformation (for more details, refer to Supplementary Section Fisher-Z transformation).

The results of this analysis are illustrated through the correlation matrix in Figure 10, showing us that the feature importance scores of most models are similar to one another. This consistency among the different models and explanation methods indicates that our evaluation framework produces reliable explanations that reflect some underlying phenomena found in the data set. We also note that for SVMs and neural networks, the correlation analysis was performed using the absolute feature importances to make them directly comparable to the importance scores of the other models (Figures 7a and 8a). The correlation between the signed feature importances for the SVM and neural network models is 0.90, which is significantly higher than the absolute feature importance correlation of 0.64.

Additionally, we observe that the SVM model's importance scores display the lowest similarity to those of the other ML models. This is likely due to the interplay of two factors: the behavior of RBF-based SVMs and the way the explanations are constructed to reflect this behavior. Regarding the nature of RBF-kernel methods, the inherent smoothness imposed by the RBF kernel means that such methods are unable to perform feature selection or weighting, as the influence of any feature depends varies smoothly with the distance to the support vectors. This necessarily causes more uniformly distributed relevance scores, as irrelevant features cannot be discarded and relevant features cannot be highlighted at prediction time. Additionally, a unique property of SVMs is their sparseness, causing their predictions to be sensitive only on the support vectors of the model. Because the support vectors lie near the decision boundary they can be interpreted as outliers of their respective classes. Entropy-based models such as neural

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

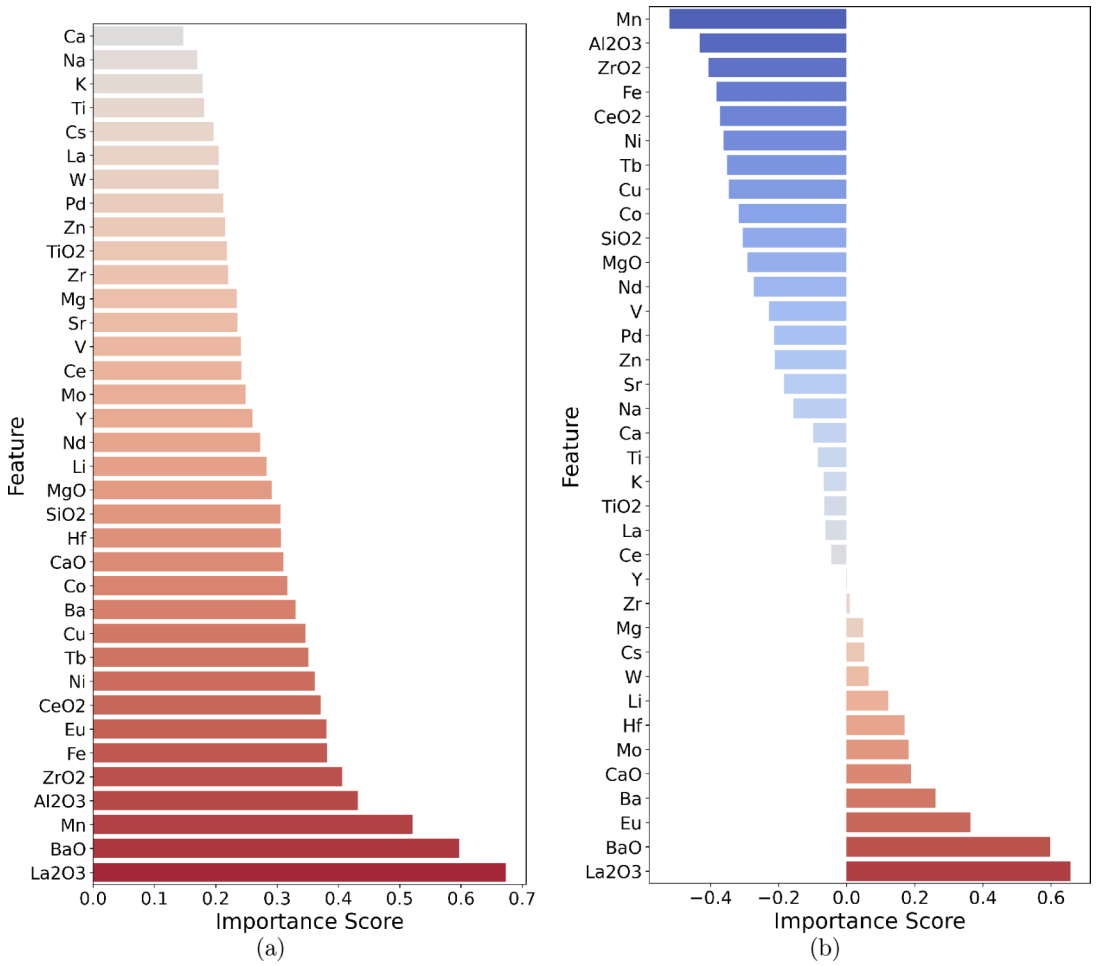


Figure 8. Feature importance analysis for SVMs via LRP based on evidence for the high-yield class. (a) Mean absolute feature importances to identify the key features independently of class-specific relevance. (b) Mean feature relevance including positive and negative relevances, disentangling the class-specific contributions of the inputs.

network classifiers and decision trees instead derive their parameters mainly based on inliers, i.e., samples that are highly representative of their respective classes. This inherent difference likely causes fundamentally different behavior at prediction time which is also interlinked with how explanations are acquired for SVMs. More specifically, the explanations obtained for SVMs via LRP are based on the distance to the support vectors of each class, which correctly reflects the behavior of the model. Thus, the sparse and local nature of SVMs along with the smooth nature of RBF kernel lead to a lower similarity of absolute feature importances when compared to other models.

Discussion of Component Contributions for High-Yield Catalyst Design. The analysis in the previous sections suggests that even if explanation methods produce only positive importance scores irrespective of class, it still does not necessarily follow that a component assigned with high importance is beneficial for creating high-yield catalysts. In fact, the component may be deemed important for classification not because it leads to the high-yield catalyst, but because its presence is likely to indicate a low-yield catalyst.

For example, none of the catalysts in the OCM data set that included Ni as a component achieved a high yield. Similarly, only one catalyst containing Mn and one containing Cu is labeled as high-yield (out of 39 and 31 samples, respectively). Therefore, Mn, Ni, and Cu are indeed key features for determining the yield of the catalyst within the context of this data set because their presence makes it very likely for a catalyst to be low-yield. This is also reflected in their high absolute importance scores across all models (Figure 9), but more importantly also in the strongly negative relevance for the signed LRP explanations (see Figures 7b and 8b).

Despite all the above, these results do not necessarily indicate that Manganese is a poor component for OCM catalysis. The explanations provided do not reveal the absolute truth but rather indicate that a specific feature strongly influences the model's classification of a sample as a low-yield catalyst within this particular data set. In previous reviews on OCM,[53,54] Manganese has been frequently cited as a favorable component, often in combination with sodium (Na) and supported by SiO_2 or MgO. However, the current data set[16] consists of 291 catalyst combinations that have been chosen randomly, and not on the basis of previous knowledge, out of a total of 36540 possible combinations. Considering this, it is very likely that the optimal combination of Mn with specific elements might be absent from the data set. Although the highly negative relevance of Mn and other components does not rule out the use of this component in producing high-yield catalysts, it certainly indicates that the component in question may have antagonistic effects when combined with other random components, making it an unattractive candidate for discovering novel high-yield catalysts.

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

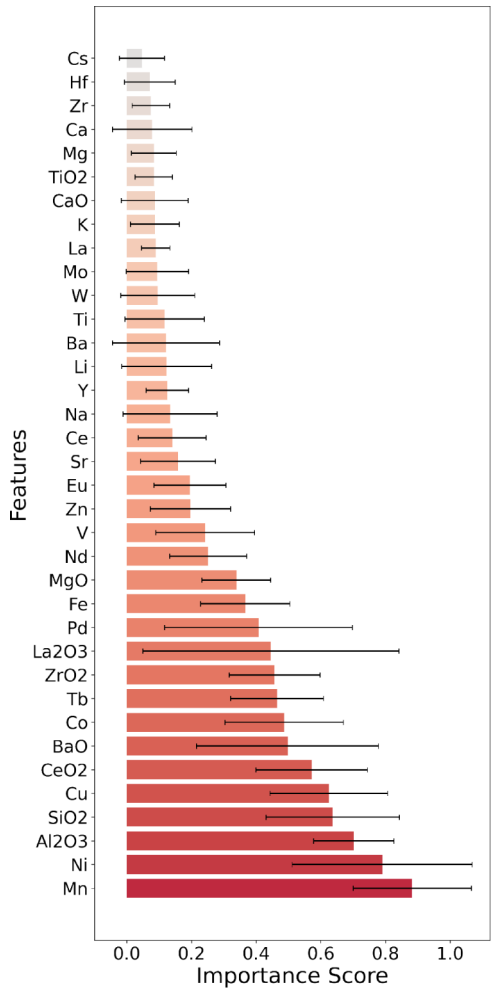


Figure 9. Average feature importance and its standard deviation (error bars) between the models (SVM, neural networks, logistic regression, and one tree-based model (random forest)).

Based on the signed LRP relevance scores, we identify two groups of the contributors to low-yield catalysts: 1) acidic supports, e.g., alumina (Al_2O_3) or zirconia (ZrO_2), and 2) Highly oxidizing metal oxides, such as Pd, Cu, Ni, Fe, Co, Ce. The supports in group 1) are shown to have a negative impact, especially when they are not neutralized by strong alkali or alkali-earth additives. We argue that this effect is caused by strong adsorption of the ethylene molecule, which is a Lewis base due to its double bond electron pair. This strong adsorption leads to further oxidation toward carbon oxides. The highly oxidizing elements in group 2) are capable of activating oxygen to strongly oxidizing species that drive the conversion of methane and/or the C_2 coupling products to carbon oxides, reducing the yield of valuable C_2 products.

On the other hand, positive importance scores are assigned to oxides (either as promoters or as supports) with a higher degree of alkalinity (BaO, CaO). This effect may arise from the improvement of ethylene desorption, which hinders its further oxidation.

Another group of elements with positive relevance are rare earth oxides, notably La and Eu. The catalytic activity of rare earth oxides in OCM reaction has been well documented in the literature,[54] with La_2O_3 being one of the best components, alongside Sm_2O_3, Gd_2O_3 and Er_2O_3. Prior research[54] shows that the Lanthanide group plays a role in activating methane as a methyl radical, which is the first step in the coupling of methane to C_2 products. An exception to this is cerium oxide, which has a negative contribution, because cerium, unlike other rare earths studied here, has a reversible valence of Ce4+/Ce3+, making it more oxidizing. This characteristic likely drives the formation of carbon oxides (total oxidation).[55] Our findings of Lanthanum oxides' positive contribution align with the literature. Unfortunately, due to the random choice of components, the other aforementioned rare earth elements are missing from the current data set.

Our analysis in this section shows that the feature importance scores assigned by our models can be related to chemical phenomena and thus can be used to guide chemists when designing novel high-yield catalysts.

Predicting Promising Catalyst Compositions via Relevance Scores. To demonstrate how feature importances can be used to generate new promising catalyst compositions, we have devised a simple generative algorithm that uses the relevances to bias the generation procedure toward catalysts that the model predicts to be high-yield.

Our algorithm is based on the procedure used to generate the data set in Nguyen et al.[16] and ensures that every sample generated is valid, i.e., it could also be generated by their random sampling procedure.

Let $\overline{R}_d$ denote the average relevance for feature d calculated over a given data set as described in eqs 18 or 19, depending on the type of model and explanation method used. Using this average relevance as an input, the generative algorithm first splits the importance scores into one set for elements and one for supports, after which the two sets of importance scores are converted into discrete probabilities using the softmax function:

$$\mathrm{softmax}(\mathbf{x}, \beta)_i = \frac{e^{\beta \mathbf{x}_i}}{\sum_i e^{\beta \mathbf{x}_i}} \tag{20}$$

where β is a temperature parameter that can be used to control the variability of the generated samples: a lower value for β will result in a more uniform distribution, while a higher value will produce a distribution where the most of the probability concentrated the few components with highest relevances scores. We also utilize two separate temperature parameters, where $\beta^{\mathcal{E}}$ is used for generating the probability distribution of the elements, while $\beta^{\mathcal{S}}$ is used for the supports.

Based on the probability distributions obtained via softmax, we first sample one support, then sample up to three elements without repetition, each time removing the last sampled element and recalculating the probabilities. Each time an element is sampled, we also include a $\frac{1}{|\mathcal{E}|}$ chance of selecting no elements, which allows our model to sample catalysts with two and one components at the same rate as the sampling method in.[16] A detailed step-by-step description can be found in Algorithm 1.

Since the feature importances are not the ground truth, but just reflect what the model determines as relevant for prediction, the candidates selected by this procedure are not guaranteed to be high-yield catalysts. However, we can verify the effectiveness of the sampling procedure by feeding the candidates generated with the feature importances back as input into the model that produced these feature importances. If the sampling procedure is effective, then the catalyst

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

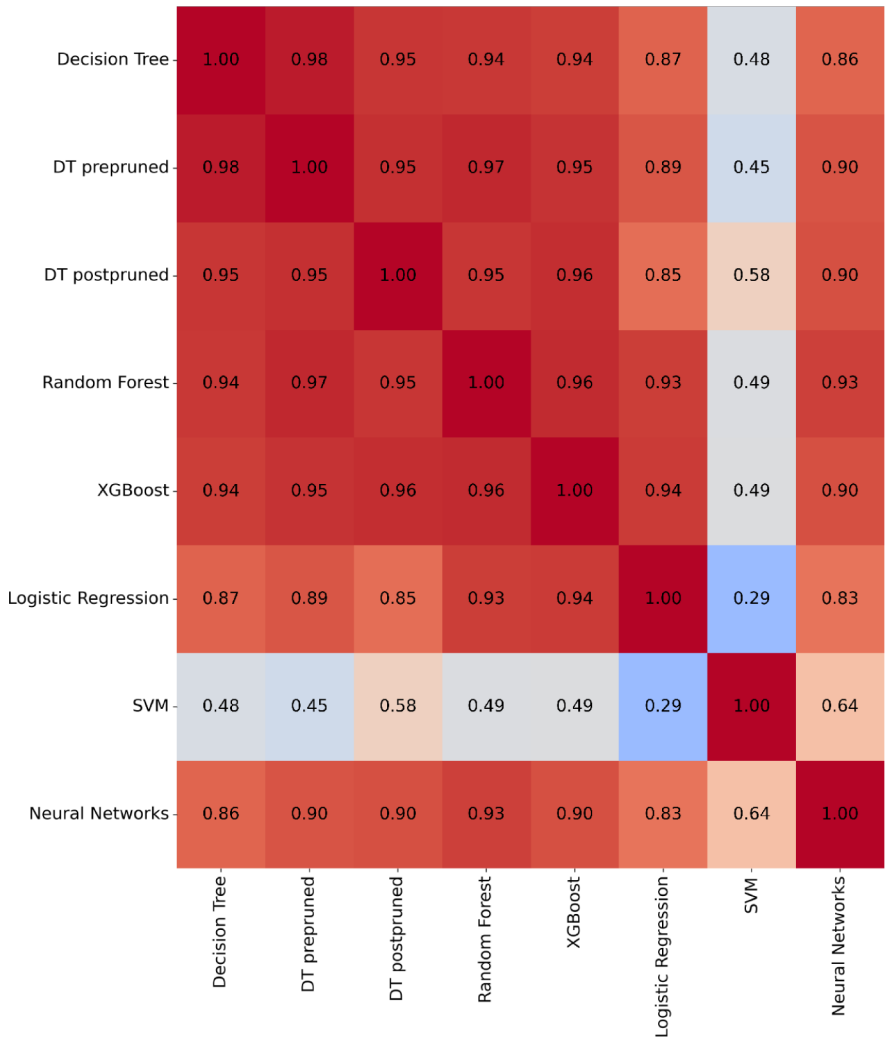


Figure 10. Pearson correlation of the feature importances between all model pairs.

Algorithm 1: A simple sampling algorithm for generating promising catalyst combinations based on feature importances provided by an explainability method.

Data:
$\bar{R}_d$ - set of feature importances
$\mathcal{E}$ - feature indices of elements
$\mathcal{S}$ - feature indices of supports
$\beta^{\mathcal{E}}$ - temperature parameter for the softmax applied on element relevances
$\beta^{\mathcal{S}}$ - temperature parameter for the softmax applied on support relevances
Result:
S^{sel} - feature index of sampled support
$E_1^{\text{sel}}, E_2^{\text{sel}}, E_3^{\text{sel}}$ - feature indices of sampled elements

$\bar{R}^{\mathcal{S}} \leftarrow \{\bar{R}_d : d \in \mathcal{S}\}$; // Select importances of support features
$\mathbf{p}^{\mathcal{S}} \leftarrow \text{softmax}(\bar{R}^{\mathcal{S}}, \beta^{\mathcal{S}})$; // Create probability distribution over supports
$S^{\text{sel}} \sim \mathbf{p}^{\mathcal{S}}$; // Sample from the probability distribution over supports
for i in $1 \ldots 3$ **do**
  $\bar{R}^{\mathcal{E}} \leftarrow \{\bar{R}_d : d \in \mathcal{E}\}$; // Select importances of element features
  $r \sim \text{uniform}(0, 1)$;
  if $r < \frac{1}{|\mathcal{E}|}$ **then** // No element is selected with a chance of $1/|\mathcal{E}|$
    $E_1^{\text{sel}} \leftarrow None$;
  else
    // Sample element and remove it from the list of indices
    $\mathbf{p}^{\mathcal{E}} \leftarrow \text{softmax}(\bar{R}^{\mathcal{E}}, \beta^{\mathcal{E}})$;
    $E_i^{\text{sel}} \sim \mathbf{p}^{\mathcal{E}}$;
    $\mathcal{E}$.remove(E_i^{sel});

candidates produced by this algorithm should be predominately classified as high-yield catalysts.

We performed these experiments for two models with different explanation methods: XGBoost using the impurity metric and neural networks using LRP. For both models, we took the average feature importances (absolute importances for XGBoost and both absolute and signed importances for neural networks) across 100 training/test splits and used them as input into the sampling algorithm to generate 1000 samples with different settings for the β parameters.

The results shown in Table 3 confirm our findings from Section Explanations Using LRP, about the additional usefulness of having explanations with class-aware feature importances.

Namely, in the case of the signed feature importances from the neural network, the proportion of generated samples classified as high-yield grows continuously as we use the temperature parameters to bias the sampling more and more toward the high-relevant features. On the other hand, using the absolute feature importances for both the neural network and XGBoost model, we observe that further biasing the sampling distribution toward features with high relevance only produces an increasing number of low-yield catalysts.

Given that most of the features with high absolute importances are also the ones with highly negative importances (see Figures 6 and 7), meaning that they mainly contribute relevance to the class of low-yield catalysts, the results in Table 3 offer further evidence about the reliability of the class-aware LRP explanations.

To showcase the type of catalyst candidates that can be produced by our algorithm, we include a list of 20 catalyst candidates generated using signed-neural network relevances as

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

Table 3. Fraction of Catalysts Generated by Our Relevance-Based Sampling Procedure That Were Classified as High-Yield by the Corresponding ML Model[abc]

Temperature	Feature importances		
parameters	NN: signed	NN: abs.	XGBoost: abs.
$\beta^{\mathcal{E}} = 10, \beta^{S} = 1$	0.38	0.17	0.31
$\beta^{\mathcal{E}} = 20, \beta^{S} = 2$	0.49	0.13	0.23
$\beta^{\mathcal{E}} = 40, \beta^{S} = 4$	0.68	0.04	0.13
$\beta^{\mathcal{E}} = 40, \beta^{S} = 4$	0.85	0.01	0.05

[a]The fractions are reported for the neural network (NN) and XGBoost models, where for the neural network the samples are generated using both the absolute feature importances (NN: abs.) and the signed class-aware feature importances (NN: signed) obtained using LRP. [b]The samples for XGBoost were generated using the absolute feature importances as obtained from the XGBoost model. [c]As the value of the beta parameters increases, we observe that the fraction of samples classified as high-yield decreases when the absolute feature importances are used, while they increase when using class-aware signed feature importances, illustrating the impact of having class-aware importances when using them to guide the development of high-yield catalysts.

input in the Supplementary Section Promising catalyst compositions obtained via XAI, along with some details on the procedure used to generate them.

To summarize, the results in this section indicate that high importance of an element or support in an ML model does not necessarily imply that including this component will produce high-yield catalysts. On the contrary, quite the opposite can be true because a high relevance alone does not give us any information about whether the feature in question predominantly contributes to the desired class. Therefore, drawing conclusions from feature importances requires using explanation methods like LRP, which can disentangle the importance and relationship of a feature to different classes.

CONCLUSION

The field of catalyst design is characterized by complex synergistic and antagonistic effects between catalyst components. This often makes high-performing catalysts difficult to discover through traditional trial-and-error methods. Machine learning, with its capacity to detect underlying patterns and complex relationships, offers great potential to accelerate catalyst discovery by identifying novel, high-performance candidates. Unfortunately, generating unbiased catalyst yield data sets is a slow and resource intensive task, resulting in small data sets where low-yield catalysts are much more dominanant than high-yield ones, making it challenging to train ML models that generalize well to high-yield outcomes.

To address the challenges posed by small, unbalanced data sets, we introduced a robust machine learning and XAI framework, incorporating resampling, cross-validation, and well-suited performance measures, as well as XAI techniques that help disentangle the positive and negative contributions of components to catalyst yield. While we have chosen to apply the framework to predicting the catalytic yield for the OCM reaction as a representative example in this case, the general design of the framework allows it to be applied for various other catalytic reactions.

Our results demonstrated that the accuracy of the various models, both with and without resampling, lies between 76 and 82%, which considering the class imbalance in the data sets, is precisely within the range of a random classifier, thereby providing misleading information about model performance. However, using the F1-score as a performance measure revealed that models with similar accuracy can have significantly different F1-scores (0.1−0.52), allowing for the identification of models who have learned to correctly distinguish the minority class of high-yield catalysts. Having this well-suited performance measure also demonstrated the positive impact of resampling, resulting in an increase of the F1-scores by at least 0.1 across all models, with the random forest model benefiting the most, with an increase of 0.42 in F1-score to reach 0.52. A notable exception to this is the SVM, which by construction is not heavily impacted by class imbalance or resampling. These findings underscore the effectiveness of our machine learning framework in enhancing model performance and reliability in catalyst yield classification.

The application of various explainable AI techniques consistently identified similar key components influencing models' decisions across different models. Notably, explanations via Layer-wise Relevance Propagation (LRP) effectively disentangled the positive and negative contributions of catalyst components. Across both SVM and neural networks, LRP explanations have highlighted the same two groups of components are the top positive contributors to high-yield catalysts: rare earth oxides (La and Eu) and alkaline earth metals with high degree of alkalinity (Ba, and Ca) as top features in driving high yield catalysts. These findings, aligning with chemical intuition and existing OCM literature, are notable given the small data set used. They further demonstrate that explainable AI can already be used to extract actionable insights from machine learning models, thereby assisting the chemist in the design of experiments for faster discovery of high-yield catalysts. As a proof of concept, we developed a sampling algorithm based on relevance scores to suggest promising catalyst compositions. The validation of this algorithm using different ML models and XAI methods once again demonstrated the importance of class-aware relevances for enabling effective ML-guided catalyst discovery.

Future research could focus on generating larger catalyst data sets that encompass diverse catalyst compositions under various process conditions. With more complex data, achieving optimal performance will require a future emphasis on enhancing the models through feature engineering and/or refining model architectures. Since our framework provided a general improvement in performance across all evaluated models, any improvements in model performance will be complementary to the advantages offered by our proposed framework, and together, these two approaches can be combined to achieve the best possible results.

Although initially designed to address the challenges of small data sets, the methods integrated into our framework will remain valuable for training and evaluating models on larger data sets. Given that unbiased catalyst data sets will still be dominated by low-yield compositions, our framework's focus on handling imbalanced data through strategies like resampling and metrics such as the F1-score, will continue to be crucial for generating reliable results.

Finally, given its unification of robust evaluation practices with interpretable explanations of complex machine learning models, we hope that the ML and XAI framework introduced in this work serve as a valuable blueprint for the community, accelerating the catalyst discovery by implementing the

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

appropriate and well-suited ML techniques tailored for experimental data, while enhancing the reliability and transparency of ML models through XAI. We believe that these techniques will make future works more interpretable, trustworthy, and impactful for catalyst design and beyond.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jpcc.4c05332.

Detailed explanation of machine learning models theory; table summarizing the ML models' performance metrics without resampling; graphical results on feature importance scores and hyper parameters of model training; single catalyst explanations using Layer-wise Relevance Propagation (LRP); the Fisher-Z transformation; the effects of periodic table group features on model explanations; a list of promising catalyst compositions identified via XAI analysis (PDF)

■ AUTHOR INFORMATION

Corresponding Authors

Parastoo Semnani − *Machine Learning Group and BASLEARN−TU Berlin/BASF Joint Lab for Machine Learning, TU Berlin, Berlin 10587, Germany; Berlin Institute for the Foundations of Learning and Data, Berlin 10587, Germany;* orcid.org/0009-0002-6607-6756; Email: p.semnani@tu-berlin.de

Klaus-Robert Müller − *Machine Learning Group, TU Berlin, Berlin 10587, Germany; Berlin Institute for the Foundations of Learning and Data, Berlin 10587, Germany; Max Planck Institute for Informatics, Saarbrücken 66123, Germany; Department of Artificial Intelligence, Korea University, Seoul 02841, South Korea;* orcid.org/0000-0002-3861-7685; Email: klaus-robert.mueller@tu-berlin.de

Authors

Mihail Bogojeski − *Machine Learning Group, TU Berlin, Berlin 10587, Germany; Berlin Institute for the Foundations of Learning and Data, Berlin 10587, Germany;* orcid.org/0000-0002-1839-7320

Florian Bley − *Machine Learning Group, TU Berlin, Berlin 10587, Germany; Berlin Institute for the Foundations of Learning and Data, Berlin 10587, Germany*

Zizheng Zhang − *Chair of Statistics and Campus Institute Data Science, Georg-August-University Göttingen, Göttingen 37073, Germany*

Qiong Wu − *Chair of Statistics and Campus Institute Data Science, Georg-August-University Göttingen, Göttingen 37073, Germany*

Thomas Kneib − *Chair of Statistics and Campus Institute Data Science, Georg-August-University Göttingen, Göttingen 37073, Germany*

Jan Herrmann − *BASF SE, Ludwigshafen 67056, Germany*

Christoph Weisser − *BASF SE, Ludwigshafen 67056, Germany*

Florina Patcas − *BASF SE, Ludwigshafen 67056, Germany*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jpcc.4c05332

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work was supported by BASLEARN, TU Berlin/BASF Joint Laboratory, cofinanced by TU Berlin and BASF SE. P.S., M.B., F.B., and K.-R.M. acknowledge support by the German Federal Ministry of Education and Research (BMBF) for BIFOLD (BIFOLD24B). K.-R.M. was partly supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the government (MSIT) (No. 2019-0-00079, Artificial Intelligence Graduate School Program, Korea University, and No. 2022-0-00984, Development of Artificial Intelligence Technology for Personalized Plug-and-Play Explanation and Verification of Explanation) and by the German Federal Ministry for Education and Research (BMBF) under Grants 01IS14013BE and 01GQ1115. C.W. acknowledges support by BASF Data and AI Academy. The authors thank Stef Lenk for illustrations (Figures 1, 2, & TOC graphic) and also Farnoush Jafari, Laure Ciernik, Rajat Kawade, and Jason Hattrick-Simpers for helpful discussions.

■ REFERENCES

(1) Graser, J.; Kauwe, S. K.; Sparks, T. D. Machine learning and energy minimization approaches for crystal structure predictions: A review and new horizons. *Chem. Mater.* **2018**, *30*, 3601−3612.

(2) Andersen, M.; Levchenko, S. V.; Scheffler, M.; Reuter, K. Beyond scaling relations for the description of catalytic materials. *ACS Catal.* **2019**, *9*, 2752−2759.

(3) Ma, S.; Liu, Z.-P. Machine learning for atomic simulation and activity prediction in heterogeneous catalysis: Current status and future. *ACS Catal.* **2020**, *10*, 13213−13226.

(4) Bogojeski, M.; Sauer, S.; Horn, F.; Müller, K.-R. Forecasting industrial aging processes with machine learning methods. *Comput. Chem. Eng.* **2021**, *144*, 107123.

(5) Ding, R.; Ding, Y.; Zhang, H.; Wang, R.; Xu, Z.; Liu, Y.; Yin, W.; Wang, J.; Li, J.; Liu, J. Applying machine learning to boost the development of high-performance membrane electrode assembly for proton exchange membrane fuel cells. *J. Mater. Chem. A* **2021**, *9*, 6841−6850.

(6) Ding, R.; Chen, Y.; Chen, P.; Wang, R.; Wang, J.; Ding, Y.; Yin, W.; Liu, Y.; Li, J.; Liu, J. Machine learning-guided discovery of underlying decisive factors and new mechanisms for the design of nonprecious metal electrocatalysts. *ACS Catal.* **2021**, *11*, 9798−9808.

(7) Esterhuizen, J. A.; Goldsmith, B. R.; Linic, S. Interpretable machine learning for knowledge generation in heterogeneous catalysis. *Nat. Catal.* **2022**, *5*, 175−184.

(8) Ishioka, S.; Fujiwara, A.; Nakanowatari, S.; Takahashi, L.; Taniike, T.; Takahashi, K. Designing catalyst descriptors for machine learning in oxidative coupling of methane. *ACS Catal.* **2022**, *12*, 11541−11546.

(9) Margraf, J. T.; Jung, H.; Scheurer, C.; Reuter, K. Exploring catalytic reaction networks with machine learning. *Nat. Catal.* **2023**, *6*, 112−121.

(10) Taniike, T.; Fujiwara, A.; Nakanowatari, S.; García-Escobar, F.; Takahashi, K. Automatic feature engineering for catalyst design using small data without prior knowledge of target catalysis. *Commun. Chem.* **2024**, *7*, 11.

(11) Vojvodic, A.; Nørskov, J. K. New design paradigm for heterogeneous catalysts. *Natl. Sci. Rev.* **2015**, *2*, 140−143.

(12) Goldsmith, B. R.; Esterhuizen, J.; Liu, J.-X.; Bartel, C. J.; Sutton, C. Machine learning for heterogeneous catalyst design and discovery. *AichE J.* **2018**, *64*, 2311−2323.

(13) Rangarajan, S. *Artificial Intelligence in Manufacturing*; Elsevier, 2024; pp. 167−204.

(14) Suzuki, K.; Toyao, T.; Maeno, Z.; Takakusagi, S.; Shimizu, K.-I.; Takigawa, I. Statistical analysis and discovery of heterogeneous

catalysts based on machine learning from diverse published data. *ChemCatchem* **2019**, *11*, 4537−4547.
(15) Wang, S.; Jiang, J. Interpretable catalysis models using machine learning with spectroscopic descriptors. *ACS Catal.* **2023**, *13*, 7428−7436.
(16) Nguyen, T. N.; Nakanowatari, S.; Nhat Tran, T. P.; Thakur, A.; Takahashi, L.; Takahashi, K.; Taniike, T. Learning catalyst design based on bias-free data set for oxidative coupling of methane. *ACS Catal.* **2021**, *11*, 1797−1809. Random catalyst OCM data by HTE.
(17) Nguyen, T. N.; Nhat, T. T. P.; Takimoto, K.; Thakur, A.; Nishimura, S.; Ohyama, J.; Miyazato, I.; Takahashi, L.; Fujima, J.; Takahashi, K.; et al. High-throughput experimentation and catalyst informatics for oxidative coupling of methane. *ACS Catal.* **2020**, *10*, 921−932.
(18) Zavyalova, U.; Holena, M.; Schlögl, R.; Baerns, M. Statistical analysis of past catalytic data on oxidative methane coupling for new insights into the composition of high-performance catalysts. *ChemCatchem* **2011**, *3*, 1935−1947.
(19) Takahashi, K.; Takahashi, L.; Le, S. D.; Kinoshita, T.; Nishimura, S.; Ohyama, J. Synthesis of Heterogeneous Catalysts in Catalyst Informatics to Bridge Experiment and High-Throughput Calculation. *J. Am. Chem. Soc.* **2022**, *144*, 15735−15744.
(20) Nishimura, S.; Le, S. D.; Miyazato, I.; Fujima, J.; Taniike, T.; Ohyama, J.; Takahashi, K. High-throughput screening and literature data-driven machine learning-assisted investigation of multi-component La 2 O 3-based catalysts for the oxidative coupling of methane. *Catal. Sci. Technol.* **2022**, *12*, 2766−2774.
(21) Toyao, T.; Maeno, Z.; Takakusagi, S.; Kamachi, T.; Takigawa, I.; Shimizu, K.-I. Machine learning for catalysis informatics: Recent applications and prospects. *ACS Catal.* **2020**, *10*, 2260−2297.
(22) Jia, X.; Lynch, A.; Huang, Y.; Danielson, M.; Lang'at, I.; Milder, A.; Ruby, A. E.; Wang, H.; Friedler, S. A.; Norquist, A. J.; et al. Anthropogenic biases in chemical reaction data hinder exploratory inorganic synthesis. *Nature* **2019**, *573*, 251−255.
(23) Brown, D. G.; Gagnon, M. M.; Bostrom, J. Understanding our love affair with p-chlorophenyl: Present day implications from historical biases of reagent selection. *J. Med. Chem.* **2015**, *58*, 2390−2405.
(24) Lapuschkin, S.; Wäldchen, S.; Binder, A.; Montavon, G.; Samek, W.; Müller, K.-R. Unmasking Clever Hans predictors and assessing what machines really learn. *Nat. Commun.* **2019**, *10*, 1096.
(25) Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; Samek, W. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLoS One* **2015**, *10*, No. e0130140.
(26) Montavon, G.; Samek, W.; Müller, K.-R. Methods for interpreting and understanding deep neural networks. *Digit. Signal Process.* **2018**, *73*, 1−15.
(27) Arrieta, A. B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion.* **2020**, *58*, 82−115.
(28) Samek, W.; Montavon, G.; Lapuschkin, S.; Anders, C. J.; Müller, K.-R. Explaining deep neural networks and beyond: A review of methods and applications. *Proc. IEEE* **2021**, *109*, 247−278.
(29) Minh, D.; Wang, H. X.; Li, Y. F.; Nguyen, T. N. Explainable artificial intelligence: A comprehensive review. In *Artificial Intelligence Review*; Springer, 2022, pp. 1−66.
(30) Saranya, A.; Subhashini, R. A systematic review of Explainable Artificial Intelligence models and applications: Recent developments and future trends. *Decis. Anal. J.* **2023**, *7*, 100230.
(31) Schmack, R.; Friedrich, A.; Kondratenko, E. V.; Polte, J.; Werwatz, A.; Kraehnert, R. A meta-analysis of catalytic literature data reveals property-performance correlations for the OCM reaction. *Nat. Commun.* **2019**, *10*, 441.
(32) He, H.; Ma, Y. *Imbalanced learning: Foundations, algorithms and applications*; John Wiley & Sons, 2013.
(33) Fernández, A.; García, S.; Galar, M.; Prati, R. C.; Krawczyk, B.; Herrera, F. *Learning from imbalanced data sets*; Springer, 2018; Vol. *10*.
(34) Tomek, I. Two Modifications of CNN. *IEEE Trans. Syst. Man. Cybern.* **1976**, *SMC-6*, 769−772.
(35) Kubat, M.; Matwin, S. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection In *Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997)*, ICML, 1997, pp 179−186.
(36) Ling, C. X.; Li, C. Data mining for direct marketing: Problems and solutions. In *Kdd*; University of Western Ontario, 1998, pp. 73−79.
(37) Drummond, C.; Holte, R. C.C4. 5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling In *Proceedings of the International Conference on Machine Learning (ICML 2003) Workshop on Learning from Imbalanced Data Sets II*, University of Alberta, 2003, pp 1−8.
(38) Liu, A.; Ghosh, J.; Martin, C.Generative Oversampling for Mining Imbalanced Datasets In *Proceedings of the 2007 International Conference on Data Mining*, DMIN, 2007, pp 66−72.
(39) Huang, C.; Li, Y.; Loy, C. C.; Tang, X. Learning Deep Representation for Imbalanced Classification In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR, 2016, pp 5375−5384.
(40) Bellinger, C.; Corizzo, R.; Japkowicz, N. Calibrated resampling for imbalanced and long-tails in deep learning In *Discovery Science: 24th International Conference*, DS, 2021, pp 242−252.
(41) Muttenthaler, L.; Vandermeulen, R. A.; Zhang, Q.; Unterthiner, T.; Müller, K. Set Learning for Accurate and Calibrated Models. In *The Twelfth International Conference on Learning Representations*, ICLR, 2024, pp 2024.
(42) Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; Kegelmeyer, W. P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321−357.
(43) Hastie, T.; Tibshirani, R.; Friedman, J. H.; Friedman, J. H. *The elements of statistical learning: Data mining, inference, and prediction*; Springer, 2009; Vol. *2*.
(44) Müller, K.-R.; Mika, S.; Rätsch, G.; Tsuda, K.; Schölkopf, B. An Introduction to Kernel-Based Learning Algorithms. *IEEE Trans. Neural Netw. Learning Syst.* **2001**, *12*, 181.
(45) Hansen, K.; Montavon, G.; Biegler, F.; Fazli, S.; Rupp, M.; Scheffler, M.; Von Lilienfeld, O. A.; Tkatchenko, A.; Muller, K.-R. Assessment and validation of machine learning methods for predicting molecular atomization energies. *J. Chem. Theory Comput.* **2013**, *9*, 3404−3419.
(46) Molnar, C.; Casalicchio, G.; Bischl, B. Interpr etable Machine Learning − A Brief History, State-of-the-Art and Challenges. In *ECML PKDD 2020 Workshops*, Springer International Publishing: Cham, 2020, pp. 417−431.
(47) Zednik, C.; Boelsen, H. Scientific Exploration and Explainable Artificial Intelligence. *Minds Mach.* **2022**, *32*, 219−239.
(48) Breiman, L. Random forests. In *Machine learning*; Springer, 2001, pp. 23−25.
(49) Breiman, L. *Classification and regression trees*; Routledge, 2017.
(50) Montavon, G.; Binder, A.; Lapuschkin, S.; Samek, W.; Müller, K.-R. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, Samek, W.; Montavon, G.; Vedaldi, A.; Hansen, L. K.; Müller, K.-R., Eds.; Springer International Publishing: Cham, 2019; pp. 193−209.
(51) Kauffmann, J.; Esders, M.; Ruff, L.; Montavon, G.; Samek, W.; Müller, K.-R. From Clustering to Cluster Explanations via Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 1926−1940.
(52) Bley, F. *Explaining Kernel Classifiers and Extensions*. Master's thesis, Technical University of Berlin, 2022.
(53) Lunsford, J. H. The catalytic oxidative coupling of methane. *Angew. Chem., Int. Ed. Engl.* **1995**, *34*, 970−980.
(54) Lee, J. S.; Oyama, S. Oxidative coupling of methane to higher hydrocarbons. *Catal. Rev.: Sci. Eng.* **1988**, *30*, 249−280.

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367

(55) Gorte, R. J. Ceria in catalysis: From automotive applications to the water−gas shift reaction. *AichE J.* **2010**, *56*, 1126−1135.

https://doi.org/10.1021/acs.jpcc.4c05332
J. Phys. Chem. C 2024, 128, 21349−21367